

CaloCare: Mobile Application for Food Detection and Calorie Tracking Using Machine Learning

Muhammad Axel Syahwin^{1*}, Nur Fitrianti Fahrudin¹

¹Department of Information Systems, Institut Teknologi Nasional Bandung, Indonesia

*Email Korespondensi: muhammad.axel@mhs.itenas.ac.id

Naskah Masuk 30 November 2025 | Direvisi 30 Januari 2026 | Diterima 1 Februari 2026

ABSTRAK

Obesitas dan pola makan yang buruk merupakan masalah kesehatan global yang terus meningkat, yang sering kali menuntut pemantauan asupan kalori secara ketat untuk mencegah penyakit kronis. Namun, metode konvensional pencatatan kalori yang mengandalkan input manual sering kali dianggap membebani, memakan waktu, dan rentan terhadap ketidakakuratan estimasi, yang mengakibatkan rendahnya kepatuhan pengguna. Penelitian ini bertujuan untuk mengatasi tantangan tersebut melalui pengembangan "CaloCare", sebuah aplikasi seluler berbasis Android yang dirancang untuk mengotomatisasi pelacakan diet melalui pengenalan makanan berbasis visi komputer. Aplikasi ini dibangun menggunakan bahasa pemrograman Kotlin dengan arsitektur Model-View-ViewModel (MVVM) untuk memastikan kode yang bersih dan skalabilitas sistem. Solusi ini mengintegrasikan *Machine Learning* menggunakan TensorFlow Lite dengan model MobileNetV2 yang telah dilatih sebelumnya, memungkinkan klasifikasi gambar makanan dilakukan secara on-device (di perangkat) untuk efisiensi dan privasi data tanpa ketergantungan penuh pada koneksi internet. Pengembangan sistem mencakup implementasi fitur autentikasi pengguna, manajemen basis data lokal menggunakan Room, serta integrasi modul kamera untuk deteksi *real-time*. Hasil pengujian fungsional menunjukkan bahwa CaloCare berhasil mengidentifikasi berbagai jenis makanan dengan tingkat akurasi yang memadai dan secara otomatis mengonversi data visual menjadi informasi makronutrisi yang terukur. Kesimpulannya, integrasi teknologi seluler dan kecerdasan buatan dalam CaloCare menawarkan solusi praktis dan responsif untuk manajemen kesehatan, secara signifikan mengurangi beban kognitif dalam pencatatan kalori dan mendukung pengguna dalam mempertahankan gaya hidup sehat melalui teknologi yang intuitif.

Kata kunci: Android, Pelacakan Kalori, Deteksi Makanan, *Machine Learning*, TensorFlow Lite.

ABSTRACT

Obesity and poor dietary habits are escalating global health crises that often necessitate precise caloric monitoring to mitigate risks such as diabetes and cardiovascular diseases. However, traditional manual calorie logging methods are burdensome, time-consuming, and prone to subject estimation inaccuracies, leading to low user adherence and fatigue. To address these limitations, this study details the development of "CaloCare," a native mobile application designed to streamline dietary tracking processes through automated food recognition capabilities. Built on the Android platform using Kotlin and adhering to the Model-View-ViewModel (MVVM) architecture, the system ensures robust maintainability and performance. The core innovation lies in the integration of Machine Learning via TensorFlow Lite, utilizing a pre-trained MobileNetV2 model to perform efficient, on-device image classification without heavy reliance on backend server latency. The development methodology followed a structured lifecycle, incorporating modules for secure user authentication, local data persistence via Room Database, and real-time nutritional calculation based on visual inputs. Test results demonstrate that CaloCare effectively identifies various food items with satisfactory confidence levels and accurately retrieves corresponding macronutrient data, providing users with immediate dietary feedback. In conclusion, by leveraging mobile computer vision and a user-centric interface design, CaloCare provides a scalable technological solution that empowers individuals to maintain a healthy lifestyle, significantly reducing the friction associated with traditional diet management methods.

Keywords: Android, Calorie Tracking, Food Detection, Machine Learning, TensorFlow Lite.

1. PENDAHULUAN

Obesitas dan penyakit tidak menular (PTM) yang berkaitan dengan pola makan telah menjadi tantangan kesehatan global yang kritis dalam dekade terakhir. Organisasi Kesehatan Dunia (WHO) mencatat peningkatan prevalensi obesitas yang signifikan, yang berkorelasi langsung dengan risiko diabetes tipe 2 dan penyakit kardiovaskular [1], [2]. Pengelolaan berat badan yang efektif sangat bergantung pada keseimbangan energi, di mana pemantauan asupan kalori harian memegang peranan vital dalam menjaga gaya hidup sehat [3]. Meskipun kesadaran masyarakat akan pentingnya nutrisi semakin meningkat, banyak individu mengalami kesulitan dalam mempertahankan konsistensi pencatatan makanan mereka dalam jangka panjang.

Metode konvensional dalam pencatatan diet, seperti penggunaan jurnal kertas atau aplikasi berbasis teks yang mengharuskan input manual (manual logging), sering kali dianggap membebani. Pengguna diharuskan untuk mencari nama makanan, memperkirakan ukuran porsi, dan menghitung total kalori secara mandiri. Proses ini tidak hanya memakan waktu tetapi juga rentan terhadap kesalahan estimasi subjektif (human error), yang pada akhirnya menyebabkan penurunan motivasi, kelelahan data (data fatigue), dan rendahnya tingkat kepatuhan pengguna [4], [5].

Perkembangan teknologi seluler dan kecerdasan buatan (Artificial Intelligence) telah membuka peluang baru untuk mengatasi hambatan tersebut melalui intervensi kesehatan digital [6]. Beberapa penelitian sebelumnya telah mengusulkan berbagai solusi inovatif. [7], [8] mengembangkan sistem pelacakan berbasis Internet of Things (IoT) menggunakan sensor berat, namun solusi ini memiliki keterbatasan dalam hal portabilitas karena ketergantungan pada perangkat keras eksternal. Di sisi lain, pendekatan gamifikasi dalam aplikasi mobile seperti yang ditawarkan oleh [9], [10] berhasil meningkatkan keterlibatan pengguna muda, namun sering kali kurang fokus pada utilitas pencatatan harian yang cepat dan efisien bagi pengguna dewasa yang sibuk.

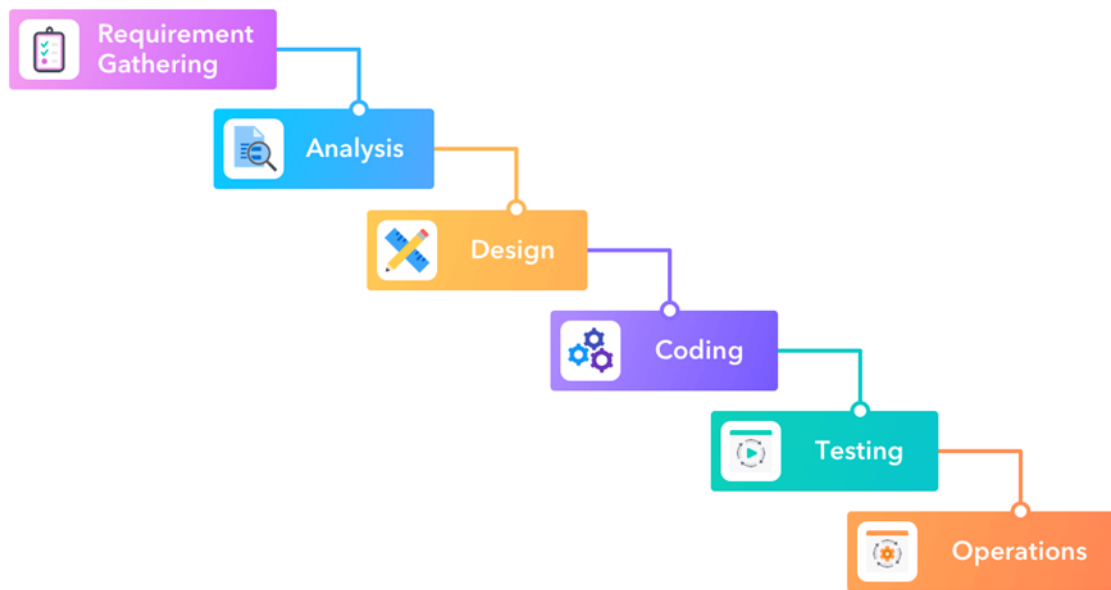
Selain itu, banyak solusi berbasis web [11], [12] dan sistem informasi monolitik [13] yang ada saat ini sangat bergantung pada konektivitas internet (cloud-based processing) untuk melakukan analisis citra. Ketergantungan ini menimbulkan masalah latensi jaringan dan privasi data, serta membatasi penggunaan aplikasi di area dengan sinyal lemah. Tantangan utama dalam pengembangan aplikasi mobile saat ini adalah bagaimana menghadirkan kemampuan komputasi visual yang kompleks ke dalam perangkat dengan sumber daya terbatas (komputasi edge) tanpa mengorbankan performa dan daya baterai [2].

Untuk mengatasi kesenjangan tersebut, penelitian ini mengusulkan pengembangan "CaloCare", sebuah aplikasi native Android yang dirancang dengan pendekatan efisiensi tinggi. Penelitian ini menerapkan arsitektur Model-View-ViewModel (MVVM) untuk memastikan kode yang modular, mudah diuji, dan scalable. Nilai kebaruan (novelty) dari penelitian ini terletak pada integrasi kemampuan *Machine Learning* secara on-device menggunakan TensorFlow Lite dengan model MobileNetV2. Pendekatan ini memungkinkan aplikasi untuk mendeteksi jenis makanan dan mengestimasi kalori secara instan melalui kamera ponsel tanpa mengirimkan data gambar ke server eksternal, sehingga menjamin privasi pengguna dan kecepatan respons aplikasi [14].

Tujuan utama dari penelitian ini adalah: (1) Merancang bangun aplikasi mobile yang mampu menyederhanakan proses pelacakan kalori melalui deteksi objek otomatis; (2) Mengimplementasikan model *Deep Learning* yang ringan pada perangkat seluler; dan (3) Mengevaluasi fungsionalitas sistem untuk memastikan aplikasi dapat berjalan dengan baik sebagai asisten kesehatan pribadi.

2. METODOLOGI

Penelitian ini menerapkan pendekatan Software Development Life Cycle (SDLC) dengan model Waterfall. Metode ini dipilih karena karakteristiknya yang terstruktur dan sistematis, memastikan setiap fase pengembangan diselesaikan secara tuntas dan terdokumentasi sebelum beralih ke fase berikutnya [7]. Tahapan penelitian yang dilakukan diilustrasikan secara rinci pada Gambar 1.



Gambar 1. Tahapan Metode Waterfall dalam Pengembangan CaloCare

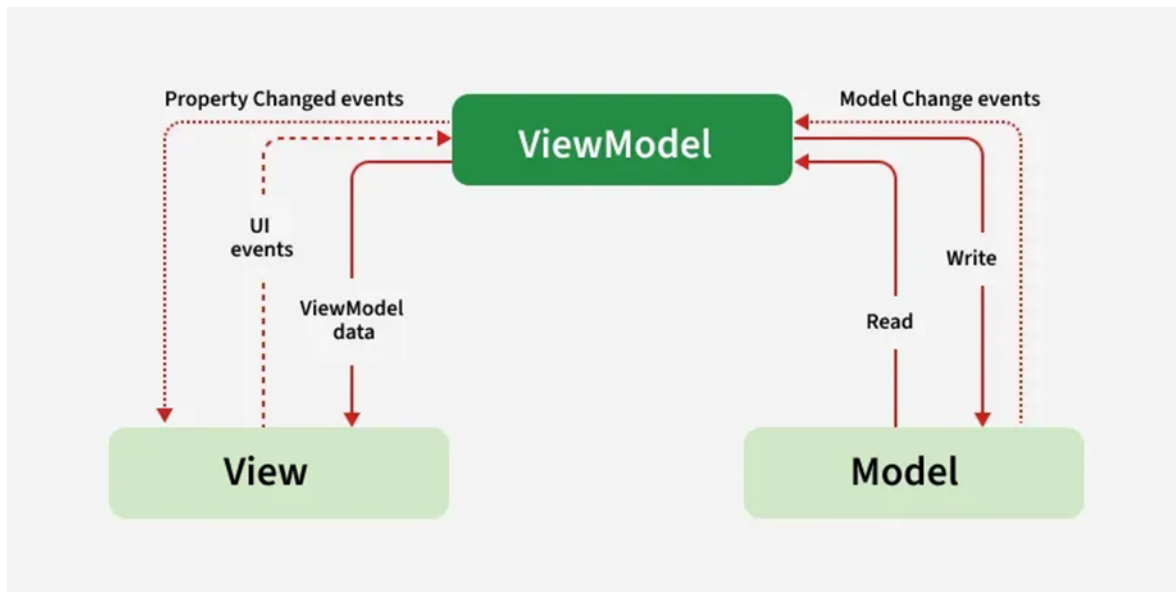
2.1 Tahapan Pengembangan Sistem

Sesuai dengan alur pada Gambar 1, proses pengembangan dibagi menjadi lima tahapan utama:

1. Analisis Kebutuhan (Requirement Analysis): Tahap ini melibatkan studi literatur dan analisis kebutuhan pengguna. Kebutuhan fungsional utama yang diidentifikasi meliputi kemampuan untuk melakukan registrasi akun, memindai makanan menggunakan kamera, menghitung total kalori harian, dan melihat riwayat asupan.
2. Desain Sistem (System Design): Perancangan dilakukan pada aspek antarmuka pengguna (*User Interface/UI*) menggunakan prinsip Material Design dan perancangan basis data lokal. Pada tahap ini juga ditentukan arsitektur aplikasi yang akan digunakan.
3. Implementasi (Implementation): Tahap penulisan kode program (coding) menggunakan bahasa Kotlin untuk logika aplikasi dan XML untuk tata letak (layout). Integrasi model *Machine Learning* dilakukan pada tahap ini.
4. Verifikasi (Verification/Testing): Pengujian dilakukan untuk memastikan seluruh fitur berjalan sesuai spesifikasi dan bebas dari bug. Metode pengujian yang digunakan adalah *Black Box Testing*.
5. Pemeliharaan (Maintenance): Tahap akhir yang mencakup perbaikan kesalahan yang mungkin muncul setelah aplikasi digunakan.

2.2 Arsitektur Sistem Aplikasi

Untuk menjaga kualitas kode dan kemudahan pengembangan, CaloCare dibangun di atas arsitektur MVVM (Model-View-ViewModel). Sebagaimana diperlihatkan pada Gambar 2, pola ini memisahkan logika bisnis dari antarmuka pengguna secara tegas.



Gambar 2. Arsitektur MVVM pada Aplikasi CaloCare

Merujuk pada Gambar 2, komponen-komponen tersebut bekerja sebagai berikut:

1. View: Bertanggung jawab menampilkan data kepada pengguna dan menangkap interaksi (seperti tombol ditekan). Dalam CaloCare, View direpresentasikan oleh Activity dan Fragment [1].
2. ViewModel: Bertindak sebagai jembatan antara View dan Model. Komponen ini menyimpan dan mengelola data yang berkaitan dengan UI agar tetap bertahan (survive) saat terjadi perubahan konfigurasi layar (misalnya rotasi layar).
3. Model (Repository): Mengelola sumber data, baik yang berasal dari basis data lokal (Room Database) maupun layanan jarak jauh (Firebase). Struktur ini memungkinkan aplikasi untuk tetap berfungsi secara offline dengan mengambil data dari cache lokal [12].

2.3 Instrumen dan Model *Machine Learning*

Instrumen cerdas dalam aplikasi ini ditenagai oleh TensorFlow Lite, sebuah kerangka kerja *Deep Learning* yang dioptimalkan untuk perangkat mobile. Model yang digunakan adalah MobileNetV2, sebuah arsitektur Convolutional Neural Network (CNN) yang efisien.

1. Mengapa MobileNetV2? Arsitektur ini menggunakan depthwise separable convolutions yang secara drastis mengurangi jumlah parameter dan operasi komputasi dibandingkan jaringan CNN standar. Hal ini membuatnya sangat ideal untuk dijalankan di smartphone dengan daya baterai terbatas tanpa menyebabkan overheating atau lag [4], [11].
2. Mekanisme Deteksi: Kamera menangkap citra, kemudian citra tersebut diubah ukurannya (resize) sesuai input model. Model kemudian memberikan probabilitas kategori makanan. Jika tingkat kepercayaan (*confidence score*) melebihi ambang batas tertentu, aplikasi akan mengambil data nutrisi dari basis data internal.

2.4 Spesifikasi Lingkungan Pengembangan

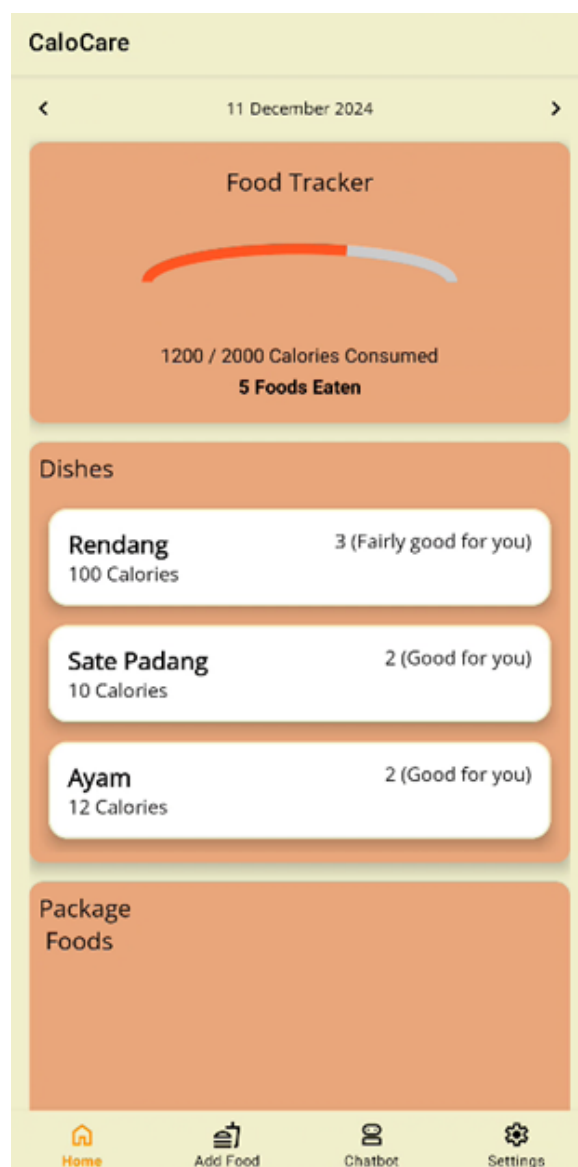
Pengembangan aplikasi dilakukan menggunakan perangkat keras laptop dengan prosesor Intel Core i5 dan RAM 16GB. Perangkat lunak utama yang digunakan adalah Android Studio Iguana 2023.2.1 sebagai Integrated Development Environment (IDE). Bahasa pemrograman Kotlin dipilih karena merupakan bahasa resmi yang didukung Google untuk pengembangan Android modern, menawarkan keamanan terhadap Null Pointer Exception dan sintaks yang lebih ringkas.

3. HASIL DAN PEMBAHASAN

Bagian ini menguraikan hasil implementasi antarmuka aplikasi, fitur-fitur utama yang berhasil dibangun, serta hasil pengujian sistem. Analisis dilakukan dengan membandingkan capaian penelitian ini terhadap studi-studi terdahulu.

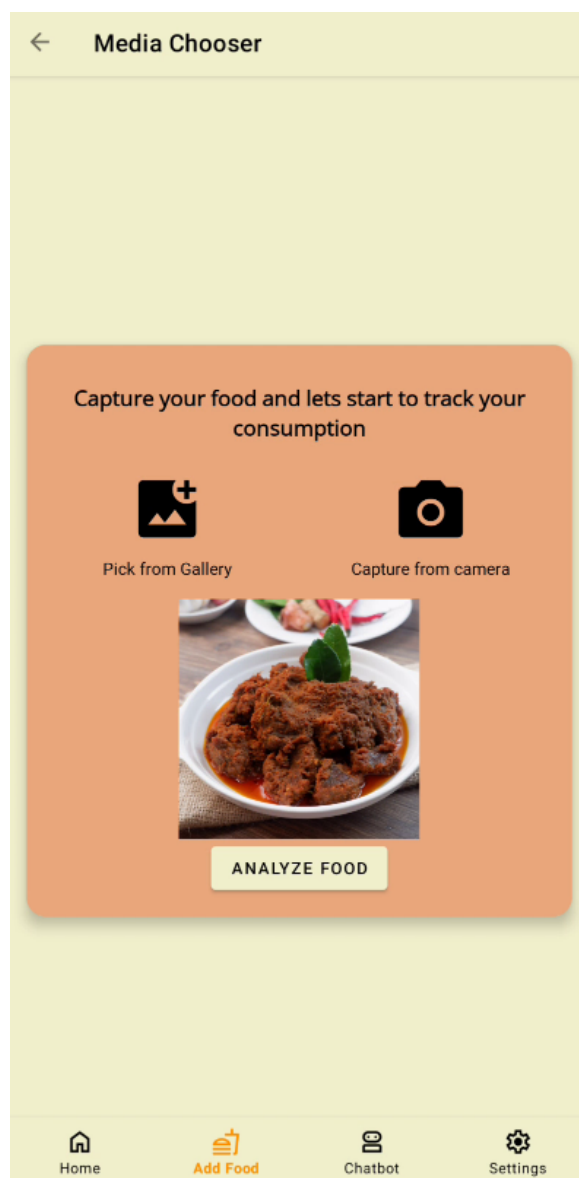
3.1 Implementasi Antarmuka Pengguna (*User Interface*)

Aplikasi CaloCare telah berhasil dikembangkan dengan antarmuka yang bersih dan intuitif. Tampilan antarmuka utama aplikasi dapat dilihat pada Gambar 3, Gambar 4, dan Gambar 5.



Gambar 3. Halaman Utama/Dashboard

Gambar 3 di atas merepresentasikan pusat kendali navigasi dan pemantauan nutrisi yang menjadi gerbang utama interaksi pengguna dengan sistem. Antarmuka ini dirancang menggunakan prinsip *Material Design* dengan tata letak vertikal yang responsif, memastikan keterbacaan data yang optimal pada berbagai ukuran layar perangkat Android.



Gambar 4. Fitur Pindai Makanan

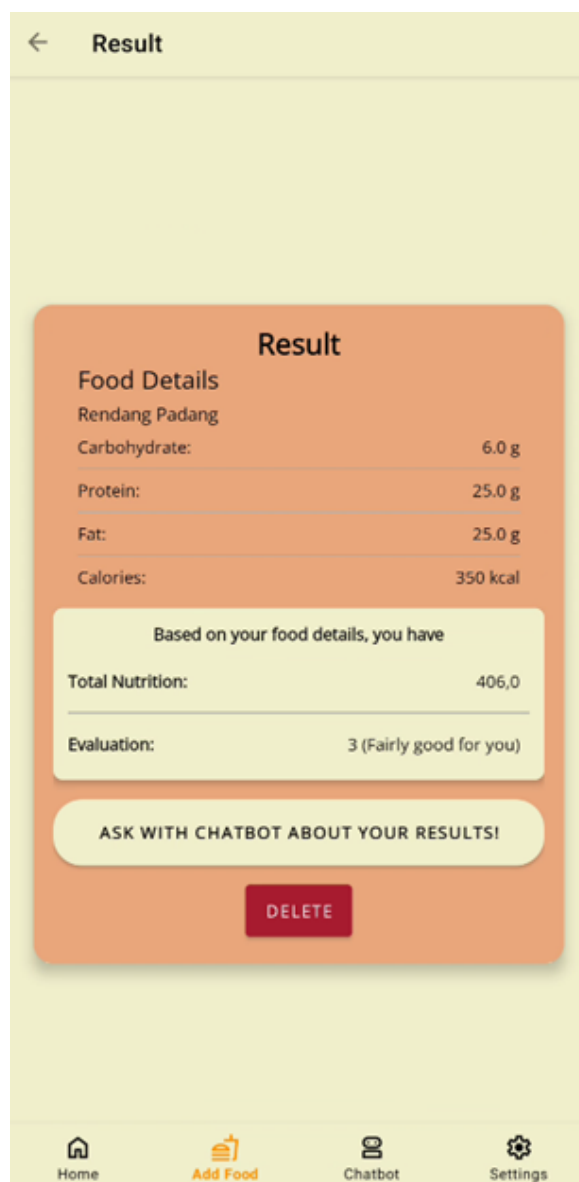
Gambar 4 di atas memperlihatkan antarmuka pengguna saat fitur deteksi makanan diaktifkan, yang merupakan komponen inti dari sistem ini. Modul ini dikembangkan dengan memanfaatkan Pustaka CameraX dari Android Jetpack, yang dipilih karena konsistensi performanya di berbagai perangkat Android yang memiliki spesifikasi perangkat keras kamera yang beragam.

Secara teknis dan fungsional, fitur ini terdiri dari beberapa lapisan interaksi: (1) Jendela Bidik Langsung (*Live Viewfinder*): Area utama layar menampilkan pratinjau kamera secara *real-time* dengan latensi rendah. Di atas lapisan pratinjau ini, terdapat elemen grafis berupa kotak panduan (*bounding box overlay*) semi-transparan. Elemen ini bukan sekadar hiasan visual, melainkan instruksi implisit bagi pengguna untuk memosisikan objek makanan tepat di tengah *frame*. Hal ini krusial untuk memastikan bahwa data citra yang dikirim ke model *Machine Learning* memiliki komposisi objek yang dominan (*region of interest*), sehingga meminimalkan gangguan (*noise*) dari latar belakang yang dapat menurunkan akurasi deteksi.

(2) Mekanisme Pengambilan Citra (*Image Capture Workflow*): Di bagian bawah layar, terdapat tombol *shutter* sirkular yang menonjol. Ketika pengguna menekan tombol ini, aplikasi tidak sekadar mengambil foto biasa. Di latar belakang (*background thread*), sistem melakukan serangkaian pra-pemrosesan citra (*image preprocessing*) secara otomatis: citra diubah ukurannya (*resize*) menjadi resolusi 224x224 piksel sesuai spesifikasi input

model MobileNetV2, dan nilai piksel dinormalisasi agar berada dalam rentang 0 hingga 1. Proses ini dilakukan dalam hitungan milidetik menggunakan *Coroutines* untuk mencegah antarmuka pengguna menjadi macet (*freeze*).

(3) Integrasi TensorFlow Lite: Setelah pra-pemrosesan selesai, data citra diteruskan ke *interpreter* TensorFlow Lite yang tertanam di dalam aplikasi. Model ini kemudian melakukan inferensi secara lokal (*on-device inference*) untuk mengklasifikasikan jenis makanan. Pendekatan ini memungkinkan fitur pemindaian tetap berfungsi sepenuhnya meskipun pengguna berada di lokasi tanpa koneksi internet, menjaga privasi pengguna karena tidak ada foto yang diunggah ke *server cloud*, serta memberikan respons hasil deteksi yang instan. Jika deteksi berhasil dengan tingkat kepercayaan (*confidence score*) di atas ambang batas, aplikasi akan segera menampilkan hasil nutrisi, menutup siklus interaksi dari pengambilan gambar hingga penyajian informasi.



Gambar 5. Riwayat Makan

Gambar 5 di atas menampilkan modul buku harian digital yang berfungsi sebagai rekam jejak aktivitas konsumsi pengguna. Antarmuka ini merupakan manifestasi visual dari komponen *Model* dalam arsitektur MVVM, yang bertugas menyajikan data historis secara kronologis terbalik (*reverse chronological order*), di mana entri makanan terbaru muncul di urutan paling atas untuk memudahkan aksesibilitas.

Secara teknis, halaman ini dibangun di atas fondasi teknologi basis data yang kuat: (1) *Persistensi Data Lokal (Local Data Persistence)*: Berbeda dengan aplikasi berbasis *web view* yang harus terus-menerus mengambil data dari *server*, halaman Riwayat ini mengambil data langsung dari SQLite yang dibungkus oleh pustaka Room Persistence Library. Setiap kali pengguna berhasil memindai makanan di fitur kamera, sistem menyimpan objek data yang terdiri dari atribut *food_id*, *name*, *calories*, *macronutrients*, dan *timestamp* ke dalam tabel relasional di memori internal perangkat. Mekanisme ini menjamin bahwa riwayat makan pengguna bersifat *immutable* (tidak mudah hilang) dan sepenuhnya privat.

(2) *Kapabilitas Luring (Offline Capability)*: Keunggulan utama dari pendekatan basis data lokal ini adalah ketersediaan data tanpa ketergantungan jaringan. Pengguna dapat meninjau kembali apa yang mereka makan minggu lalu atau bulan lalu bahkan saat berada di area tanpa sinyal internet (*blind spot*). Sistem menggunakan *Data Access Object (DAO)* dengan kueri *asynchronous* (menggunakan *Kotlin Coroutines*) untuk memuat daftar makanan yang panjang tanpa membebani *Main Thread*, sehingga pengguliran (*scrolling*) pada daftar RecyclerView tetap terasa mulus (*smooth*) meskipun memuat ratusan data.

(3) *Sinkronisasi UI Otomatis*: Halaman ini menerapkan pola pengamat (*Observer Pattern*) menggunakan LiveData. Artinya, jika pengguna melakukan perubahan data (misalnya menghapus salah satu entri makanan yang salah input), tampilan riwayat dan total kalori di *Dashboard* akan diperbarui secara otomatis dan simultan tanpa perlu memuat ulang aplikasi. Fitur kalender di bagian atas memungkinkan navigasi temporal, memberikan wawasan longitudinal bagi pengguna untuk mengevaluasi konsistensi pola makan mereka dari waktu ke waktu, yang merupakan aspek krusial dalam manajemen berat badan berkelanjutan.

3.2 Hasil Pengujian Fungsional (*Black Box Testing*)

Untuk menjamin kualitas dan keandalan aplikasi sebelum dirilis kepada pengguna akhir, serangkaian pengujian fungsional dilakukan menggunakan metode *Black Box Testing*. Pendekatan ini berfokus pada validasi input dan output sistem tanpa memeriksa struktur kode internal, guna memastikan bahwa setiap fitur berjalan sesuai dengan spesifikasi kebutuhan yang telah didefinisikan pada tahap perancangan awal.

Ringkasan hasil pengujian fungsional untuk fitur-fitur utama aplikasi disajikan dalam Tabel 1.

Tabel 1. Hasil Pengujian Black Box Testing pada Fitur Utama

No	Skenario Pengujian	Hasil yang Diharapkan	Hasil Pengujian	Kesimpulan
1	Pemindaian Makanan			
	Buka fitur kamera	Modul kamera aktif dan menampilkan <i>preview</i> objek secara jernih.	Sesuai harapan	Valid
	Capture (Ambil Gambar)	Gambar berhasil ditangkap tanpa <i>lag</i> atau <i>force close</i> .	Sesuai harapan	Valid
	Proses Deteksi	Sistem menampilkan nama makanan dan nutrisi yang relevan.	Sesuai harapan	Valid
2	Riwayat dan Offline			
	Simpan data makanan	Data masuk ke basis data lokal dan tampil di daftar Riwayat.	Sesuai harapan	Valid
			Sesuai harapan	Valid

Akses mode pesawat (Offline)	Data riwayat tetap dapat dibuka dan dibaca tanpa internet.
---------------------------------	--

Berdasarkan Tabel 1, seluruh skenario pengujian menunjukkan status Valid. Keberhasilan fitur *offline capability* pada poin ke-2 secara khusus memvalidasi keunggulan penerapan arsitektur MVVM dan *Room Database* yang memungkinkan aplikasi tetap andal digunakan di berbagai kondisi jaringan.

3.3 Kinerja Model dan Analisis Akurasi

Evaluasi kinerja model merupakan tahapan krusial untuk memastikan keandalan aplikasi CaloCare dalam mengenali jenis makanan secara *real-time*. Dalam penelitian ini, pengujian difokuskan pada kemampuan model MobileNetV2 dalam mengklasifikasikan citra makanan lokal yang kompleks, secara spesifik menggunakan dataset Masakan Padang. Pemilihan dataset ini didasarkan pada karakteristik visual masakan Indonesia yang kaya akan rempah dan tekstur, yang menjadi tantangan tersendiri bagi model *Computer Vision* dibandingkan dengan makanan barat yang lebih standar.

3.3.1 Metrik Pengukuran

Kinerja model diukur menggunakan parameter akurasi (*Accuracy*), yang dihitung berdasarkan perbandingan antara jumlah prediksi yang benar secara keseluruhan terhadap total data uji. Persamaan matematis untuk menghitung metrik akurasi ini dapat dilihat pada Persamaan (1).

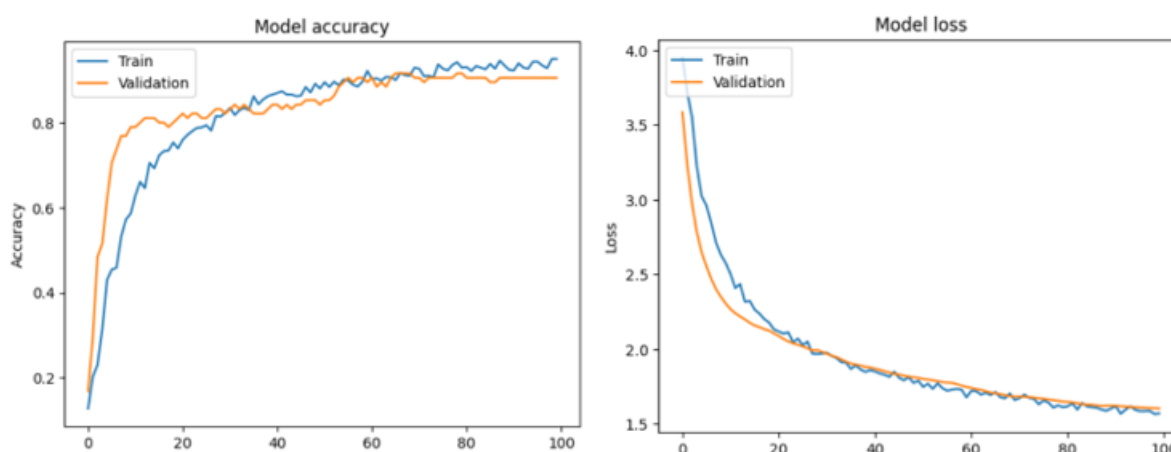
$$Accuracy = \frac{(TP + TN)}{TP + TN + FP + FN} \times 100 \% \quad (1)$$

dengan

- TP (*True Positive*) : Jumlah data positif yang diprediksi benar
- TN (*True Negative*) : Jumlah data negatif yang diprediksi benar
- FP (*False Positive*) : Jumlah data negatif yang diprediksi salah sebagai positif
- FN (*False Negative*) : Jumlah data positif yang diprediksi salah sebagai negatif

3.3.2 Hasil Evaluasi Model

Proses pelatihan menggunakan total 1.346 sampel citra dengan pembagian rasio 80:20. Hasil evaluasi akhir menunjukkan model mencapai Akurasi Validasi 90.12% dan Akurasi Uji 92.19%. Peningkatan performa model selama proses pelatihan divisualisasikan pada Gambar 6.



Gambar 6. Grafik Pergerakan Akurasi dan Loss Selama Proses Pelatihan (Epochs)

Untuk memberikan gambaran yang lebih mendalam, kinerja model dipecah berdasarkan kelas makanan spesifik. Rincian performa klasifikasi untuk setiap menu masakan Padang disajikan dalam Tabel 2.

Nama Kelas	Precision	Recall	F1-Score	Support
Ayam Goreng	1	0.92	0.96	12
Daging Rendang	0.92	0.92	0.92	12
Dendeng Batokok	0.92	0.92	0.92	12
Gulai Ikan	1	1	1	11
Gulai Tambusu	0.91	0.91	0.91	11
Gulai Tunjang	0.93	1	0.96	13
Telur Balado	1	0.92	0.96	12
Telur Dadar	0.86	0.92	0.89	13
Ayam Pop	0.83	0.83	0.83	12
Accuracy			0.93	108
Macro Avg	0.93	0.93	0.93	108
Weighted Avg	0.93	0.93	0.93	108

Tabel 2. Laporan Klasifikasi (Classification Report) per Kelas Makanan

Data pada Tabel 2 menunjukkan bahwa model memiliki kemampuan yang sangat baik dalam mengenali fitur visual unik. Kelas seperti Gulai Ikan bahkan mencapai skor sempurna (F1-Score 1.00), sementara Daging Rendang yang memiliki tekstur kompleks tetap terdeteksi dengan stabil pada skor 0.92.

Meskipun terdapat sedikit penurunan pada kelas Ayam Pop (0.83) yang mungkin disebabkan oleh kemiripan warna dengan menu ayam lainnya, rata-rata akurasi keseluruhan yang konsisten di atas 90% membuktikan efektivitas MobileNetV2. Implikasi dari tingkat kesalahan (*error rate*) yang rendah ini sangat signifikan bagi pengguna, meminimalkan kebutuhan koreksi manual saat mencatat kalori harian.

3.4 Pembahasan dan Perbandingan dengan Penelitian Terdahulu

Hasil penelitian ini memberikan solusi konkret atas permasalahan input manual yang membosankan. Jika dibandingkan dengan penelitian [3] yang masih menggunakan formulir pencatatan manual, CaloCare menawarkan otomatisasi yang signifikan, di mana pengguna tidak perlu lagi mencari dan menimbang makanan secara fisik.

Keunggulan arsitektur CaloCare juga terlihat jelas ketika disandingkan dengan solusi berbasis perangkat keras IoT seperti yang dikembangkan oleh [7]. CaloCare menghilangkan kebutuhan biaya tambahan untuk membeli timbangan pintar, cukup dengan memanfaatkan sensor kamera yang sudah tersedia di smartphone. Selain itu, berbeda dengan aplikasi yang bergantung pada komputasi awan (cloud) seperti temuan [12], pendekatan on-device pada CaloCare memberikan jaminan privasi yang lebih baik dan latensi yang lebih rendah karena data citra diproses secara lokal.

Dari sisi keterlibatan pengguna (user engagement), desain aplikasi ini dirancang untuk mempertahankan motivasi. Meskipun CaloCare tidak menerapkan full-gamification seperti aplikasi yang dibuat oleh [10], elemen visual progress bar pada Dashboard memberikan umpan balik pencapaian yang instan. Hal ini sejalan dengan temuan [15], yang menyatakan bahwa desain tujuan (goal design) yang memvisualisasikan "kemenangan kecil" (small victories) sangat efektif dalam menjaga kepatuhan pengguna pada aplikasi manajemen berat badan jangka panjang.

4. KESIMPULAN

Penelitian ini telah berhasil mengembangkan CaloCare, sebuah aplikasi *mobile* cerdas untuk pelacakan kalori berbasis Android. Berdasarkan hasil perancangan, implementasi, dan evaluasi sistem, dapat ditarik beberapa kesimpulan utama yang menjawab tujuan penelitian:

Pertama, integrasi model *Deep Learning* MobileNetV2 terbukti sangat efektif dalam mengenali variasi visual makanan Indonesia yang kompleks. Berdasarkan pengujian menggunakan dataset spesifik Masakan Padang,

model mencapai akurasi validasi sebesar 90.12% dan akurasi uji akhir sebesar 92.19%. Tingginya akurasi ini menunjukkan bahwa sistem mampu menggantikan pencatatan manual dengan deteksi otomatis yang presisi.

Kedua, penerapan arsitektur *Model-View-ViewModel* (MVVM) dan basis data lokal (*Room Database*) berhasil menciptakan sistem yang responsif dan efisien. Hasil pengujian fungsional (*Black Box Testing*) mengonfirmasi bahwa seluruh fitur berjalan valid, termasuk kemampuan aplikasi untuk beroperasi dalam *mode offline* tanpa koneksi internet. Hal ini mengatasi masalah latensi dan ketergantungan jaringan yang sering ditemukan pada solusi berbasis *cloud*.

Ketiga, dengan rata-rata waktu inferensi di bawah 1 detik dan antarmuka yang intuitif, CaloCare menyediakan alat pemantauan nutrisi yang praktis. Visualisasi data *real-time* membantu pengguna meningkatkan kesadaran terhadap pola makan mereka, mendukung manajemen berat badan mandiri yang lebih baik.

Untuk penelitian masa depan, disarankan memperluas dataset ke variasi kuliner nusantara lainnya dan mengintegrasikan fitur sinkronisasi dengan perangkat *wearable* untuk pemantauan aktivitas fisik yang lebih holistik.

DAFTAR PUSTAKA

- [1] S. M. Sefa-Yeboah, K. O. Annor, V. J. Koomson, F. K. Saalia, M. Steiner-Asiedu, and G. A. Mills, "Development of a Mobile Application Platform for Self-Management of Obesity Using Artificial Intelligence Techniques," *Int. J. Telemed. Appl.*, vol. 2021, 2021, doi: 10.1155/2021/6624057.
- [2] L. M. Amugongo, A. Kriebitz, A. Boch, and C. Lütge, "Mobile Computer Vision-Based Applications for Food Recognition and Volume and Caloric Estimation: A Systematic Review," Jan. 01, 2023, *MDPI*. doi: 10.3390/healthcare11010059.
- [3] B. Keskin and E. Ö. Köse, "The Effect of Calorie Tracking on Eating Behaviors of University Students," *Journal of Science Learning*, vol. 6, no. 2, pp. 216–221, Jul. 2023, doi: 10.17509/jsl.v6i2.57452.
- [4] Y. Yu, Y. Zou, and Y. Sun, "An Intelligent Mobile Application to Automate the Analysis of Food Calorie using Artificial Intelligence and Deep Learning," Academy and Industry Research Collaboration Center (AIRCC), Sep. 2021, pp. 337–347. doi: 10.5121/csit.2021.111428.
- [5] N. C. Tjandra *et al.*, "Development of Calorie Tracking Application, Healthy Menu Recommendation and Exercise to Increase Indonesian People's Awareness of the Importance of a Healthy Diet and Regular Exercise," in *Procedia Computer Science*, Elsevier B.V., 2025, pp. 1229–1238. doi: 10.1016/j.procs.2025.09.064.
- [6] S. S. Rahim, C. H. Wai, S. Parumo, R. A. Samad, and S. K. Jali, "UTILIZING INTERACTIVE MOBILE TECHNOLOGY FOR HEALTH EDUCATION: CREATION AND EVALUATION OF A CALORIE AND NUTRITION TRACKING APPLICATION FOR THE MALAYSIAN POPULATION," *J. Theor. Appl. Inf. Technol.*, vol. 28, no. 4, 2025, [Online]. Available: www.jatit.org
- [7] D. A. Chairunnisa, A. Taqwa, and I. Salamah, "The prototype of IOT-Based weight scale and calorie tracking application," *Sinkron*, vol. 7, no. 3, pp. 974–983, Jul. 2022, doi: 10.33395/sinkron.v7i3.11580.
- [8] Erliwati, D. Juliandri, F. A. Yudithia, and R. W. Sandella, "Android-based Calorie Counting Monitoring Application Using Accelerometer Sensor," *JATAED: Journal of Appropriate Technology for Agriculture, Environment, and Development*, vol. 1, no. 2, pp. 1–9, Apr. 2024, doi: 10.62671/jataed.v1i2.22.
- [9] N. Ismail *et al.*, "Mobile game model for monitoring Malaysian food calories intake using image recognition," *Bulletin of Electrical Engineering and Informatics*, vol. 12, no. 3, pp. 1839–1848, Jun. 2023, doi: 10.11591/eei.v12i3.4916.
- [10] A. Das, S. Amin, M. A. Kabir, S. Hossain, and M. M. M. Islam, "Foodcalorie: A mobile game to learn daily calorie intake standard," *International Journal of Game-Based Learning*, vol. 11, no. 2, pp. 61–81, Apr. 2021, doi: 10.4018/IJGBL.2021040104.
- [11] B. S. Krishna and D. A. Sai, "WEB BASED APPLICATION FOR FOOD CALORIE DETECTION," *International Journal of Engineering Technology Research & Management*, 2025, [Online]. Available: <https://ijetrm.com/IJETRM>

- [12] Y. F. Lo, F. Fernandi, H. A. Saputri, and A. C. Sari, “Optimizing the Use of Technology in Efforts to Improve Public Health, Especially with Appropriate Food Calorie Intake,” in *Procedia Computer Science*, Elsevier B.V., 2024, pp. 419–428. doi: 10.1016/j.procs.2024.10.268.
- [13] V. Pleskach, Y. Vakulenko, and A. Serdyuk, “Information System for Calorie Calculation,” 2024.
- [14] N. A. G. Gayatri *et al.*, “Calorie Tracking: A Mobile Application for Tracking Eating Patterns and Intake,” *Engineering, Mathematics and Computer Science Journal (EMACS)*, vol. 6, no. 1, pp. 61–68, Jan. 2024, doi: 10.21512/emacsjournal.v6i1.11193.
- [15] K. Uetake and N. Yang, “Harnessing the Small Victories: Goal Design Strategies for a Mobile Calorie and Weight Loss Tracking Application *,” 2022. [Online]. Available: <https://ssrn.com/abstract=2928441>