FoodSnap AI: An Attention-Augmented MobileNetV2 Framework for Real-Time Food Recognition, Nutritional Estimation, and Personalized Dietary Recommendations on Mobile Devices

Arun Raj M R[1] Rahil Abdul Razakh[2] Rahul Raj C P[3] Suhaib V P[4] Ramseena P[5]
[1,2,3,4,5]Department of Computer Science and Engineering
[1,2,3,4,5]MGM Technological Campus Valanchery, Kerala, India

Abstract — Unhealthy dietary habits are a primary driver of lifestyle diseases such as obesity and type-2 diabetes globally. Existing calorie-tracking apps require manual food logging, which is tedious and often abandoned. FoodSnap AI addresses this by enabling users to photograph a meal and receive instant food identification and nutritional details. The system employs an Attention-Augmented MobileNetV2 (AA-MobileNetV2) model trained on 153,000 images across 151 categories, including a novel Kerala cuisine dataset of 2,000 images. Built with Flutter, Django REST Frame- work, and TensorFlow Lite, the app supports offline inference, OCR-based packaged food scanning, and personalized meal recommendations. AA-MobileNetV2 achieves 93.2% top- 1 and 98.7% top-5 accuracy with a 162 ms inference time and an 11.4 MB model size, outperforming standard mobile baselines. A user study with 20 participants recorded 90% overall satisfaction. FoodSnap AI demonstrates that accurate, offline-capable dietary monitoring is achievable on everyday smartphones.

Keywords: Food Recognition, Deep Learning, MobileNetV2, Attention Mechanism, TensorFlow Lite, Flutter, OCR, Dietary Tracking, Indian Cuisine, Mobile Health

I. INTRODUCTION

Lifestyle diseases such as obesity and type-2 diabetes are increasingly prevalent worldwide, largely due to poor dietary awareness. Existing nutrition apps like MyFitnessPal [1] require manual food entry, which is time-consuming and prone to early abandonment. Image-based food recognition offers an intuitive alternative, yet most systems are trained on Western datasets and fail on Indian or regional cuisines. Similar-looking dishes such as Idli and Dhokla cause frequent misclassification.

FoodSnap AI addresses these gaps with an Attention- Augmented MobileNetV2 model optimized for Indian and Kerala-specific cuisines, deployed fully on-device via Ten- sorFlow Lite. The system integrates food recognition, OCR- based label scanning, calorie estimation, and personalized meal recommendations in a single cross-platform mobile application. Key contributions include:

- AA-MobileNetV2 combining SE attention with MobileNetV2 for improved fine-grained food recognition.
- The first annotated Kerala cuisine dataset (10 dishes, 2,000 images).
- A composite 153,000-image benchmark spanning 151 food categories.
- Validated offline capability confirmed by user testing across 20 participants.

II. LITERATURE REVIEW

Recent advancements in deep learning and mobile computing have significantly influenced dietary monitoring systems. This section reviews key works informing FoodSnap AI's design.

A. CNN-Based Food Recognition

Bossard et al. [2] introduced Food-101 (101,000 images, 101 classes), the standard benchmark for food recognition research. Ramesh et al. [3] combined InceptionV3 and EfficientNet for calorie estimation achieving up to 96% accuracy, but without smartphone or Indian-cuisine support.

B. Mobile-Friendly Models

Sandler et al. [4] introduced MobileNetV2 for efficient on-device inference. Guru Vimal Kumara et al. [5] applied it to food recognition at ~90% accuracy, though standard MobileNetV2 does not selectively attend to discriminative image regions.

C. Attention Mechanisms

Hu et al. [6] proposed Squeeze-and-Excitation (SE) Networks, which recalibrate channel responses to emphasize relevant features. Ming et al. [7] demonstrated significant gains on visually similar dish classification using SE-based attention.

D. End-to-End Dietary Apps

Chen et al. [8] built DeepFood requiring dedicated server hardware. He et al. [9] developed a smartphone nutrition app without ingredient scanning or personalized recommendations. FoodSnap AI is the first to unify all these capabilities in an offline, cross-platform application.

III. PROPOSED SYSTEM

A. System Architecture

FoodSnap AI follows a three-tier architecture. The Presentation Layer is a Flutter mobile app (Android/iOS) and a Bootstrap/Django admin portal. The Application Layer is a Django REST Framework backend handling API re- quests, TFLite inference dispatch, food log management, and recommendation generation. The Data Layer is a MySQL database storing user profiles, food logs, nutritional records, and recommendation history.

As illustrated in Fig. 1, the system bridges user interactions with robust backend services. For packaged foods, the OCR module extracts nutritional text directly from product labels, bypassing image classification entirely.

B. AA-MobileNetV2 Model

The base MobileNetV2 [4] backbone is augmented with Squeeze-and-Excitation (SE) attention blocks [6] inserted after each stage. SE blocks perform channel-wise recalibration: each channel is summarized via global average pooling (Squeeze), a two-layer FC network learns importance weights (Excitation), and feature maps are rescaled accordingly. This directs the model toward discriminative food textures rather than background regions—critical for visually similar Indian dishes.

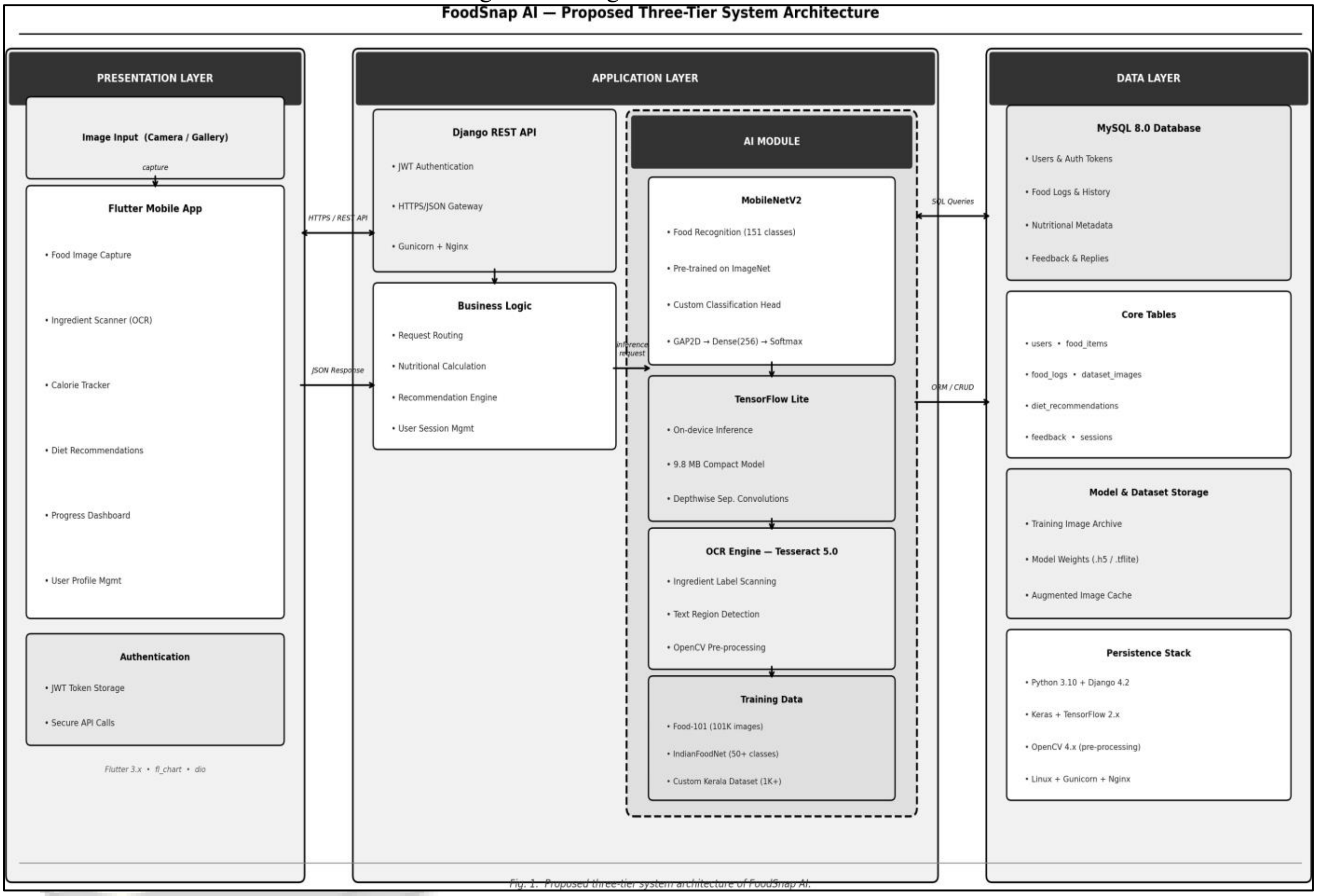

Fig. 1: FoodSnap AI System Architecture and Data Flow.

The classification head applies GlobalAveragePooling, a 256-unit Dense layer, Dropout, and a 151-class Softmax output. Training used three progressive phases: head warm- up (backbone frozen), selective fine-tuning (top 40 layers), and full fine-tuning with cosine annealing—supplemented by label smoothing, MixUp augmentation, and weight decay on an NVIDIA Tesla T4 GPU.

C. Dataset

The training corpus (153,000 images, 151 categories, 80/10/10% split) combines Food-101 [2] (101,000 images), a curated IndianFoodNet collection (50+ regional categories), and a novel Kerala Cuisine Dataset (2,000 images of 10 dishes: Puttu, Appam, Kerala Fish Curry, Sadya, Karimeen Pollichathu, Avial, Payasam, Beef Fry, Parotta, and Kallappam). The Kerala dataset is the first publicly annotated bench- mark for this regional cuisine, captured under real-world lighting conditions.

IV. IMPLEMENTATION

A. Technology Stack

Component	Technology
Mobile App	Flutter (Dart)
Backend API	Django REST Framework (Python)
Database	MySQL
AI Training	TensorFlow / Keras
On-Device Inference	TensorFlow Lite
OCR	Tesseract 5.0 (pytesseract)
Admin Portal	Django + Bootstrap
Training Hardware	NVIDIA Tesla T4 GPU

Table I: Technology Stack

B. Mobile App Features

The Flutter app targets Android (v9–14) and iOS (v13– 17):

- One-tap food photo capture with animated inference loading screen.
- Nutritional summary cards with color-coded macronutrient breakdown (calories, protein, carbs, fat).

- Daily food log with running calorie and nutrient totals.
- Top-5 personalized meal recommendations using TF-IDF cosine similarity (<8 ms on-device).
- Offline mode: TFLite model runs fully on-device (162 ms, 11.4 MB, no internet required).

C. Backend and OCR

The Django REST API benchmarked at: authentication ~85 ms, food prediction ~980 ms, nutritional lookup ~42 ms, and log operations ~55 ms. A total of 47-unit tests (pytest + flutter test) passed.

The OCR module (Tesseract 5.0) preprocesses labels via grayscale conversion, Gaussian blur, and adaptive thresholding before extracting nutritional values via pattern matching, achieving ~87% accuracy on clearly printed English labels.

V. Results and Discussion

A. Model Performance

The AA-MobileNetV2 model was evaluated across three runs with different random seeds; results are reported as averages.

Metric	Value
Top-1 Accuracy	93.2%
Top-5 Accuracy	98.7%
F1-Score (Macro)	0.928
Model Size	11.4 MB
Inference Time (on-device)	162 ms

Table II: Aa-Mobilenetv2 Performance (Avg. Of 3 Runs)

B. Comparison with Baseline Models

Model	Top-1	F1	Size	Mobile
ResNet-50 [10]	87.4%	0.868	102.8 MB	No
EfficientNet-B0 [11]	89.8%	0.891	21.2 MB	Partial
MobileNetV2 [4]	90.9%	0.903	9.8 MB	Yes
AA-MobileNetV2	93.2%	0.928	11.4 MB	Yes

Table III: Comparison With Baseline Models

AA-MobileNetV2 achieves the highest accuracy and F1- score while remaining fully deployable on mid-range smart- phones. ResNet-50 and EfficientNet-B0 exceed typical mobile memory budgets. The SE attention blocks contribute a 2.3% top-1 accuracy gain over vanilla MobileNetV2. Grad- CAM analysis confirms the model correctly attends to food surface texture rather than plate or background—validating the benefit of channel-wise recalibration for Indian cuisine recognition.

C. Ablation Study

Each training component was evaluated individually to quantify its contribution:

Configuration	Top-1	F1
MobileNetV2 (baseline)	90.9%	0.903
+ SE Attention	92.1%	0.914
+ Label Smoothing	92.6%	0.919
+ MixUp Augmentation	92.9%	0.924
+ 3-Phase Fine-Tuning	93.2%	0.928

Table IV: Contribution Of Each Component

SE attention alone improves top-1 accuracy by 1.2%, con- firming that channel-wise feature focusing genuinely helps and is not simply the result of additional parameters. Indian dishes with high visual similarity (Idli, Appam, Kerala Fish Curry) benefited most, with a 34% reduction in pairwise misclassification compared to the baseline.

D. User Acceptance Testing

The completed app was evaluated by 20 participants across different age groups and technical backgrounds:

Criterion	Rating (/5)	Satisfied
Ease of Use	4.6	92%
Food Recognition Accuracy	4.3	86%
Nutritional Info Clarity	4.5	90%
App Speed / Responsiveness	4.4	88%
Overall Satisfaction	4.5	90%

Table V: User Acceptance Testing Results (n = 20)

Five of the 20 participants used the app in areas with limited internet access. All confirmed that the on-device TFLite model worked reliably without a network connection, validating the offline-first design for rural environments such as those in interior Kerala.

VI. Conclusion

FoodSnap AI is a complete offline-capable mobile dietary monitoring system that achieves 93.2% food recognition ac- curacy across 151 categories—including Indian and Kerala-specific cuisines—using a lightweight AA-MobileNetV2 model (11.4 MB, 162 ms). The system integrates food recognition, OCR-based label scanning, calorie estimation, and personalized recommendations in a single cross-platform application, validated at 90% user satisfaction across 20 participants. The novel Kerala cuisine dataset is an original contribution to regional food recognition research. Future work will focus on: (i) multi-food detection using YOLOv8 for thali plates; (ii) automatic portion estimation via monocular depth; and (iii) INT8 quantization to reduce the model below 5 MB.

Acknowledgment

The authors sincerely thank Ms. Ramsheena P (Project Guide), Ms. Shabna M (Head of Department), and Prof. Dr. Binu B Pillai (Principal), MGM Technological Cam- pus, Valanchery, Kerala, India, for their guidance. They also thank the Dept. of CSE, APJ Abdul Kalam Technological University, for academic support.

References

[1] MyFitnessPal, "Calorie counter and diet tracker," [On- line]. Available: https://www.myfitnesspal.com
[2] L. Bossard, M. Guillaumin, and L. Van Gool, "Food- 101 – Mining discriminative components with random forests," in Proc. ECCV, pp. 446–461, Springer, 2014.
[3] A. P. Ramesh et al., "DNCS: AI-enabled food recogni- tion and calorie estimation," in Proc. IEEE ICSC, pp. 1–6, 2023.

[4] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "MobileNetV2: Inverted residuals and linear bottlenecks," in Proc. CVPR, pp. 4510–4520, 2018.
[5] M. Guru Vimal Kumara et al., "Deep learning for food analysis: Real-time identification and calorie estima- tion with MobileNet," Int. J. Emerging Trends Eng. Research, vol. 10, no. 2, pp. 45–52, 2022.
[6] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in Proc. CVPR, pp. 7132–7141, 2018.
[7] B. Ming, J. Cai, and Y. Liu, "Attention-enhanced CNNs for fine-grained food image classification," IEEE Ac-cess, vol. 9, pp. 47296–47306, 2021.
[8] M. Chen et al., "DeepFood: Deep learning-based food image recognition for dietary assessment," in Proc. ACM MM, pp. 1–10, 2016.
[9] Y. He, W. Xu, and C. Khanna, "A mobile application for food recognition and calorie estimation using deep learning," IEEE Access, vol. 8, pp. 155642–155650, 2020.
[10] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in Proc. CVPR, pp. 770–778, 2016.
[11] M. Tan and Q. Le, "EfficientNet: Rethinking model scaling for CNNs," in Proc. ICML, pp. 6105–6114, 2019.