What is there in an Indian *Thali?*

Yash Arora
IIIT Hyderabad
Hyderabad, India
yasharora102@gmail.com

Aditya Arun
IIIT Hyderabad
Hyderabad, India
adityaarun1@gmail.com

C.V. Jawahar
IIIT Hyderabad
Hyderabad, India
jawahar@iiit.ac.in

Abstract

Automated dietary monitoring solutions face significant challenges when dealing with culturally diverse, multi-dish meals, where traditional single-item recognition approaches fail to capture the complexity of real-world eating patterns. Most existing computer vision systems are tailored to western foods and struggle with the overlapping textures, varied presentations, and cultural specificity of dishes like Indian *Thalis*, which contain 5–10 distinct food items per plate. We present Food Scanner, a novel, end-to-end pipeline with retraining-free segmentation & prototype-based classification, plus a lightweight trainable weight-regression head for automated nutrition estimation of multi-dish meals from a single image. Our approach requires no class-specific segmentation or classification retraining, enabling rapid adaptation to new dishes and cuisines. The pipeline integrates zero-shot segmentation, embedding-based prototype classification, a lightweight weight regression head, and nutrition computation to transform an Indian *thali* into per-dish calorie and macronutrient breakdowns. To enable this study, we contribute two datasets: a multi-view Indian *Thali* dataset of 796 plates (7,900 images) covering 50 dishes (with dense plate-level masks), and a weight estimation dataset of 267 plates (1,394 images) covering 41 dishes (with gram-level weight annotations). Systematic ablation studies show that our method achieves high accuracy while maintaining real-time performance. By combining zero-shot capabilities with a modular design, Food Scanner offers a scalable, culturally adaptable solution that can be deployed across diverse food environments without any additional training. The code will be available here.

CCS Concepts

• **Computing methodologies** → **Image segmentation**; **Neural networks**; • **Applied computing** → Health informatics.

Keywords

food computing, image segmentation, weight estimation, dietary monitoring, Indian Thali dataset

ACM Reference Format:
Yash Arora, Aditya Arun, and C.V. Jawahar. 2025. What is there in an Indian *Thali?. In Indian Conference on Computer Vision, Graphics, and Image Processing (ICVGIP 2025), December 17–20, 2025, Mandi, India. ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3774521.3774594*

ACM ISBN 979-8-4007-1930-1/25/12
https://doi.org/10.1145/3774521.3774594

Figure 1: From *thali* to Nutritional Summary: Our Food Analysis Pipeline. This figure illustrates the components of our proposed solution: accurate multi-item segmentation (b), precise weight estimation (c), and comprehensive nutritional summarization (d) from a multi-dish Indian thali image (a).

1 Introduction

In recent years, the intersection of artificial intelligence and food analysis, termed food computing, has evolved into a vibrant field, propelled by breakthroughs in computer vision and deep learning [25, 41]. This transforms dietary assessment from subjective manual methods (e.g., food diaries or recall-based surveys [35]) to objective, image-based solutions that extract nutritional information directly from meal photographs [21, 29, 33]. These advances enhance dietary monitoring and enable large-scale applications in public health, food service, and nutritional epidemiology [8].

Food analysis through computer vision constitutes a multi-stage pipeline comprising food item recognition, pixel-wise segmentation, volume or weight estimation, and finally, nutritional inference [36, 40]. Among these, food segmentation and weight estimation form the technical backbone of reliable calorie computation. Segmentation enables the localization and isolation of individual food items, while weight prediction transforms visual signals into actionable metrics such as portion size and energy content [23]. This sequential dependency implies that inaccuracies in early-stage segmentation can severely impair downstream estimation tasks.

Earlier efforts in food AI have largely focused on single-item classification or object detection, often leveraging CNN architectures trained on large public datasets like Food-11 [32] and Food-101 [3]. However, these datasets frequently lack the granularity required

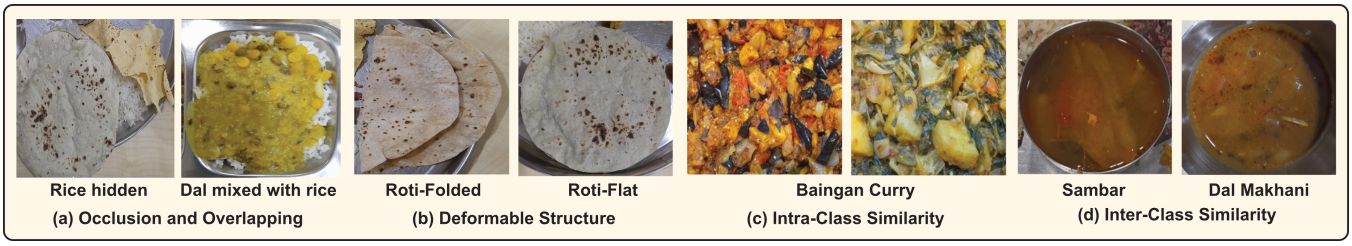

Figure 2: Visual Challenges in Automated Analysis of Indian Cuisine. This figure illustrates key complexities that hinder food analysis, including: (a) severe occlusion and overlapping of items; (b) deformable structures like folded vs. flat rotis; (c) high intra-class similarity within a single dish category like baingan-curry; and (d) high inter-class similarity between different dishes like Sambar and Dal-makhani.

for segmentation or weight annotation, particularly in complex, multi-item meal settings. Moreover, the preponderance of Western cuisines in these datasets creates a notable gap in culturally diverse food representations, especially those resembling real-world dining conditions in regions like India.

While modern supervised architectures achieve outstanding performance on curated datasets such as our Indian *Thali* Dataset, these gains often come at the cost of heavy model training, substantial annotated data requirements, and limited adaptability to new dish categories. In dynamic environments, such as university mess halls or corporate cafeterias, where lighting, plating, and menu composition can change daily, retraining or fine-tuning such models is resource-intensive and logistically challenging. This motivates our design of a modular, training-free segmentation and classification pipeline that trades a small drop in segmentation accuracy for vastly greater scalability and operational flexibility.

Among global culinary traditions, the Indian *thali* represents one of the most visually complex and computationally challenging meal formats. A typical *thali* comprises 5-10 distinct dishes arranged in close proximity on a single plate, including rice preparations, *dals*, rotis, chutneys, pickles, and sweets. These items often have similar colors and textures—such as *dal* vs. *sambar* or cabbage vs. *poriyal*—complicating instance-level segmentation.

To address these limitations, we introduce a multi-view Indian *Thali* Dataset comprising around 796 plates covering 50 unique dishes, with a total of 7900 high-resolution images captured from 7-10 viewpoints per plate. Collected in real-world dining hall conditions, each image is annotated with precise pixel-level masks. This segmentation and labelling resource enables training and evaluation of both conventional and foundation models for food segmentation under realistic, non-studio lighting and arrangement conditions. Unlike existing food datasets [5, 10, 16, 19, 26, 37], which often focus on single-item, Western-style meals in controlled environments, our dataset reflects variability and complexity inherent to Indian dining and fills a major gap in food-computing resources.

To extend beyond segmentation and enable downstream nutritional analysis, we introduce an independent Weight Estimation Dataset (WED) consisting around 267 plates, covering 41 unique dishes, and totalling 1394 images. For each plate, we recorded gram-level weight measurements of individual food regions using a scale. Although these measurements come from a separate set rather than being directly paired with our segmentation masks, they support supervised learning of visual-to-weight estimation models. The resulting dataset enables models to infer real-world portion sizes from 2D image cues—a critical step in calorie-estimation pipelines.

To demonstrate the practical utility of our approach, we validate a fully automated, modular pipeline in a university mess setting for calorie estimation from plated *thali* images (see Section 4.1). The system begins by generating class-agnostic segmentation masks using GroundedSAM2 (GSAM2) [28], followed by region classification via visual embedding comparisons using the Perception Encoder [2]. Each labelled food region is then passed to our FusionWeightNet architecture to capture class-specific weight patterns and spatial relationships within each food region. Calorie values are subsequently computed using a predefined lookup table, producing a full nutritional profile from a single image without requiring any manual input at inference time. This system performs calorie estimation with minimal user intervention, illustrating how segmentation, classification, and regression can be integrated into a scalable pipeline for culturally informed, computer-vision-based dietary monitoring.

Accurate segmentation and weighing of individual *thali* components enable diverse practical and commercial applications. In dietary monitoring, such systems can replace manual logging with automated estimates, while in commercial settings like cafeterias, they enable automated checkout and nutritional transparency. Furthermore, large-scale deployment supports public health research by objectively tracking population-level nutritional trends.

In summary, our work makes the following key contributions:

- We validate a modular calorie estimation system, which integrates segmentation, a prototype-based classification module, and our trained regression network to deliver end-to-end nutritional feedback This pipeline is designed for practical, daily deployability in dynamic environments like cafeterias and hospitals, addressing a critical gap where no comparable solutions for Indian cuisine currently exist.
- We introduce a multi-view Indian *Thali* dataset with pixel-level segmentation masks for 50 food categories, captured under authentic dining conditions.
- We provide a food weight dataset, linking each segmented dish region to precise gram-level scale measurements, enabling visual-to-mass model training and evaluation.
- We benchmark a range of supervised segmentation models—including convolutional architectures (FCN [22], SegNeXt [12], DeepLabv3+ [4]), transformer-based methods

(SegFormer [38], Mask2Former [7]), and foundation model (ClipSeg [24])—on the proposed Indian *Thali* dataset.
- We develop a ROI-based weight prediction architecture built on ResNet-50 [15], exploring modality combinations (RGB-only, depth-only, RGB+depth), attention mechanisms (CBAM and self-attention), and geometric features (ROI area and mean depth statistics) through systematic ablation studies for robust, class-specific weight estimation.

The remainder of this paper is organized as follows. Section 2 reviews prior work and key concepts in food segmentation and weight estimation. Section 3 outlines the creation of the Indian *Thali* Dataset (ITD) and the associated segmentation pipeline. Section 4 introduces the Weight Estimation Dataset (WED) and describes the proposed model. Section 4.1 details the deployed weight and calorie estimation system. Finally, Section 6 summarizes the main findings and discusses future research directions.

2 Preliminaries

2.1 AI for Food

Segmentation models have achieved impressive performance on natural images, but food presents its own challenges, as illustrated in Figure 2. In food computing, automated pipelines now leverage advances in computer vision and deep learning to translate raw images into actionable dietary insights [25, 41]. Core tasks include food recognition and classification, where systems have moved from handcrafted features to CNN-based architectures that learn hierarchical representations directly from data [13, 39], and food segmentation, which produces pixel-level masks critical for isolating individual dishes prior to quantitative analysis.

Architectures have progressed from FCNs [22] and U-Net [30] to instance-level models like DeepLabv3+ [4], efficient designs like SegNeXt [12], and transformers like Mask2Former [7] and SegFormer [38], which model long-range dependencies. Concurrently, foundation models like CLIPSeg [24] introduced zero-shot capabilities, allowing mask generation without extensive per-task retraining.

In food recognition and detection, pretrained networks such as MobileNetV2 [31] and ResNet [15] have demonstrated strong accuracy on benchmarks like Food-101 [3] and VireoFood172 [18], often using transfer learning and data augmentation to overcome limited training data. For segmentation, architectures have progressed from FCNs [22] and U-Net [30] to more powerful instance-level models like Mask R-CNN [14] and DeepLabv3+ [4], and most recently to promptable, foundation models such as SAM [20, 27] that can generate masks without per-task training. Weight and volume estimation studies—e.g., Nutrition5K [34] and MetaFood3D [6] have shown that integrating depth sensors or multi-view captures with RGB features significantly improves the fidelity of mass predictions, paving the way for end-to-end calorie estimation systems.

A rich ecosystem of food image datasets underpins these advances. Collections such as Food-11 [32] and Food-101 [3] focus on classification but lack segmentation or weight annotations. Nutrition5K [34] introduced depth scans alongside RGB for volume estimation, but does not provide per-item masks. MetaFood3D [6] supplies textured 3D meshes, RGB-D videos, segmentation masks, and weight labels, yet remains small in scale. Indian-cuisine datasets are sparse: existing collections target classification or detection, with few offering pixel-level masks or weight measurements. This leaves a gap for multi-item, multi-view datasets with true weight annotations in authentic dining environments.

Figure 3: Representative Examples of Images and Pixel-Level Annotations. This figure showcases the diverse range of multi-view Indian thali images, complete with precise pixel-level masks for each distinct dish. These detailed annotations support fine-grained segmentation and serve as ground truth for accurate food analysis.

2.2 Challenges with Indian Food

Indian *thalis* pose a formidable challenge for computer vision systems due to their inherent complexity and diversity. Unlike single-item meals, a *thali* assembles 5-10 distinct dishes—ranging from *dals, rice, rotis, pickles, and sweets*—often arranged in close proximity on the same plate. Many dishes share similar color palettes (for example, dal and sambar) and exhibit subtle texture differences, making pixel-wise discrimination inherently difficult. Moreover, small components such as chutneys or pickles can occupy only a tiny fraction of the image, increasing the risk of false negatives or merged segments under standard segmentation models.

Real-world dining hall conditions further exacerbate these challenges. *Thalis* are photographed under varying illumination and viewpoints, introducing shadows and perspective distortions. Crowd-sourced and wild-setting Indian food datasets report high diversity in image quality, background clutter, and camera distances, yet none address pixel-exact segmentation suited to multi-item plates. This variability demands models that are robust to changes in illumination and viewpoint, and capable of fine-grained boundary recovery when adjacent dishes share textures or hues.

The deformable, non-rigid nature of many Indian dishes adds a layer of difficulty. Unlike rigid objects, food items change shape when served—curries pool, breads fold, and rice grains scatter, so learning a fixed volumetric or shape template is infeasible. Combined with intra-class variability from ingredient substitutions (e.g., using toor dal versus masoor dal in dal preparations) and portion-size fluctuations, these factors drive a need for extensive, culturally specific data and advanced architectures that can handle both inter-class similarity and intra-class diversity without overfitting.

3 Dishes on the Thali Plates

3.1 Indian Thali Dataset (ITD)

The diversity and complexity of an Indian *thali*, where multiple dishes coexist on a single plate, pose unique challenges. To address

this critical gap in culturally-specific food computing resources, we introduce the Indian Thali Dataset (ITD), the first large-scale, multi-view dataset with dense, pixel-level annotations for these complex meals. Unlike Western meals with one or two items, a *thali* may contain curries, rice, rotis, pickles, and salads arranged in close proximity. Therefore, a dedicated segmentation dataset is essential to disentangle these overlapping regions, accurately localize each component, and support downstream tasks, such as weight estimation or nutritional analysis. By providing high-quality pixel-level masks for each food item, the Indian *Thali* Dataset fills a critical gap in existing food vision benchmarks and lays the groundwork for models tuned to the visual intricacies of Indian cuisine.

To construct the dataset, we collaborated with our university mess. Photographs were taken over a two-week period, covering multiple menus at 7 to 10 different viewing angles for each plate. Controlled lighting and a standardized camera-to-plate distance minimized shadows and scale variations. Expert annotators then outlined each food item, ensuring that the boundaries between similar-colored dishes (eg., *dal* and *sambar*) were captured precisely.

The dataset encapsulates substantial real-world variability in *thali* presentation: portion sizes differ from one plate to another based on individual dietary preferences and serving norms; garnishing styles and minor ingredient substitutions introduce compositional diversity; and fluctuations in lighting and camera angles yield modest color and texture variations. Furthermore, by photographing each dish from multiple viewpoints, models trained on this data can generalize more effectively to the diverse conditions encountered in everyday dining environments.

To isolate individual components for subsequent weight and calorie estimation, we trained multiple semantic segmentation models on the Indian *Thali* Dataset (ITD). All models were initialized from publicly available checkpoints and fine-tuned using an 80%–20% train–validation split to ensure consistent comparison.

We benchmarked three categories of models: convolutional networks (FCN, DeepLabv3+, SegNeXt), transformer-based architectures (Mask2Former, SegFormer), and the vision–language foundation model CLIPSeg. Each model was trained to predict per-pixel class labels for all food categories in the dataset, addressing challenges such as visually similar adjacent dishes, occlusion, and non-uniform lighting conditions.

For CLIPSeg, we performed domain-specific fine-tuning to leverage its text-prompt–driven segmentation capability for culturally specific food items. Each class label in the ITD was paired with a concise natural-language prompt (e.g., *"dal"*, *"roti"*, *"sambar"*), allowing the model to exploit its pretrained vision–language alignment while adapting to the textures and plating styles of Indian cuisine.

Segmentation outputs were stored as class-specific binary masks and served as direct inputs to the classification and weight estimation modules described in later sections.

3.2 Results and Analysis

Table 1 presents the performance evaluation of all methods on our Indian *Thali* Dataset using three complementary metrics that capture different aspects of segmentation quality. A detailed breakdown of per-class Intersection-over-Union (IoU) metrics is provided in the Supplementary (Table 2) to offer more granular insight into

Figure 4: Label instance counts for the Indian *Thali* dataset, comparing the distribution of classes in the training (teal) and testing (pink) sets. The data is sorted by frequency, revealing a pronounced long-tail distribution where staple items appear frequently while many distinct dishes have significantly fewer examples.

model performance on individual food categories. Overall Pixel Accuracy (OPA) measures the fraction of correctly classified pixels across all classes:

$$\text{OPA} = \frac{\text{Correct Pixels}}{\text{Total Pixels}} \tag{1}$$

providing a holistic view of segmentation accuracy. Class-Agnostic Pixel Accuracy (CAPA) evaluates the model's ability to distinguish foreground objects from background regardless of specific class predictions:

$$\text{CAPA} = \frac{\text{Matched Foreground Pixels}}{\text{Total GT Foreground Pixels}} \tag{2}$$

focusing purely on object detection capability. Instance-Level Accuracy (ILA) measures the fraction of ground-truth instances correctly identified and labeled:

$$\text{ILA} = \frac{\text{\# of GT instances correctly identified (IoU} \geq \text{thr)}}{\text{Total \# of GT instances}} \tag{3}$$

emphasizing object-level understanding rather than pixel-level accuracy.

For fairness, the architectures for each segmentation method were selected such that the number of parameters was kept approximately the same, ensuring that performance differences arise primarily from architectural design rather than model size.

Among CNN-based architectures, performance varies significantly across metrics. FCN (ResNet-50 backbone) achieves the lowest performance with 84.46% OPA and 76.49% ILA, though it maintains relatively high CAPA at 95.83%. DeepLabv3+ (ResNet-50) shows substantial improvements, reaching 89.29% OPA and 88.01% ILA while maintaining 96.16% CAPA. SegNeXt-L demonstrates exceptional performance across all metrics, achieving the highest scores in the CNN category with 97.15% OPA, 97.32% CAPA, and 99.53% ILA, illustrating the effectiveness of multi-scale convolutional attention for food segmentation.

Transformer-based models consistently deliver strong performance. Mask2Former with ResNet-50 backbone achieves 97.35% OPA, 97.54% CAPA, and 99.52% ILA, while SegFormer-B3 performs comparably with 97.08% OPA, 97.27% CAPA, and 99.54% ILA. The

Mask2Former variant with Swin-T backbone achieves the best transformer performance at 97.28% OPA, 97.45% CAPA, and 99.58% ILA, demonstrating the benefits of hierarchical vision transformers for dense prediction tasks.

Model Family	Type	Method	CAPA	OPA	ILA
CNN Based	Trainable	FCN(R50) [22]	95.83	84.46	76.49
		DeeplabV3+(R50) [4]	96.16	89.29	88.01
		SegNeXt (L) [12]	**97.32**	**97.15**	**99.53**
Transformer Based	Trainable	Mask2Former (R50) [7]	97.54	97.35	99.52
		SegFormer (B3) [38]	97.27	97.08	99.54
		Mask2Former (SwinT) [7]	**97.45**	**97.28**	**99.58**
Foundation Models	Trainable	ClipSeg (Zero-Shot) [24]	98.23	60.42	29.83
		ClipSeg (Fine-Tuned) [24]	**95.59**	**95.54**	**99.30**
Proposed (Sec. 5)	Training Free	GSAM2 + PE (Global)	96.85	88.09	84.51
		GSAM2 + PE (Per-Date)	**96.85**	**93.06**	**91.16**

Table 1: Performance comparison of segmentation methods on the Indian *Thali* Dataset. While supervised trainable CNN and transformer models attain the highest segmentation metrics, our *training-free* Grounded-SAM-2 (GSAM2) + Perception Encoder (PE) pipeline delivers competitive results without retraining, enabling rapid adaptation to new dishes and scalable deployment in real-world food environments.

Foundation models reveal interesting domain adaptation patterns. Zero-shot CLIPSeg shows a stark performance gap, achieving only 60.42% OPA and 29.83% ILA despite maintaining reasonable CAPA at 98.23%. This disparity highlights the model's ability to detect object boundaries while struggling with accurate classification and complete instance segmentation in the food domain. After fine-tuning, CLIPSeg performance dramatically improves to 95.54% OPA, 95.59% CAPA, and 99.30% ILA, demonstrating the benefits of domain-specific adaptation for large-scale pre-trained models.

Our training-free, scalable strategy stands out in scenarios where data annotation is expensive and frequent retraining is impractical. While supervised trainable CNN and transformer models attain the highest segmentation scores, transformers in particular exceeding 97% OPA and 99% ILA, they depend on large, labeled datasets and model updates whenever new dishes appear. In contrast, our *zero-shot* Grounded-SAM-2 (GSAM2) + Perception Encoder (PE) pipeline achieves 93.06% OPA and 91.16% ILA without retraining. This small trade-off in peak accuracy is made for a significant gain in operational flexibility. Methods like SegFormer, while powerful, are impractical for dynamic environments such as cafeterias where daily menu changes would necessitate constant re-annotation and retraining. Our training-free design eliminates this bottleneck, making it a viable real-world solution. Compared to zero-shot CLIPSeg, which reaches only 60.42% OPA and 29.83% ILA, our method delivers a substantial boost in both pixel- and instance-level accuracy, while narrowing the gap to state-of-the-art transformer performance despite no fine-tuning. This is enabled by leveraging generic mask proposals from Grounded-SAM-2 (GSAM2) and matching them to class prototypes in an embedding space. The approach supports seamless, on-the-fly integration of new dishes by simply adding their image prototypes, eliminating annotation-heavy retraining cycles. This makes it particularly suited for culturally diverse and evolving food environments where rapid adaptation and scalability are critical. Moreover, the use of per-date prototypes allows the system to naturally handle intra-class variability, eg, the same dish may exhibit noticeably different colors or textures on another day due to changes in cooking time, ingredient sourcing, spice proportions, or garnish, as well as inter-class similarity between visually alike dishes such as sambar and dal. By capturing the specific presentation and preparation style of a given day's menu, the system reduces confusion between classes and maintains robust accuracy without requiring retraining.

4 Weight from Images

4.1 Weight Estimation Dataset (WED)

Estimating the mass and nutrient content of a plated meal directly from an image addresses a core challenge in dietary monitoring: what and how much food is consumed. To solve the "how much" part of this challenge for Indian cuisine, we developed the Weight Estimation Dataset (WED), a novel resource providing precise, gram-level weight annotations for individual thali items captured in a real-world dining environment. In our setup, the task was framed as follows: given one of the *thali* images, can a model predict the weight (in grams) of each component, and by extension, its caloric and macronutrient values, without any manual input? Solving this problem end-to-end would enable users to capture a quick photo of their meal and receive an immediate breakdown of the portion sizes and nutritional intake.

Several prior efforts have tackled visual weight or volume estimation, most notably the Nutrition5K [34] dataset, which pairs Western-style lunch meals with depth scans and scale measurements, and benchmarks focused on single-food items in highly controlled settings. While these resources are valuable, they often depend on specialized capture rigs, limited dish variety, or uniform backgrounds that do not reflect typical dining conditions in the wild. In contrast, our Food Weight Dataset is grounded in authentic Indian *thali* service, featuring multiple dishes per plate and a true-to-life presentation from a university mess.

We adhered to the same multi-angle and controlled lighting protocols established for our segmentation images to assemble the dataset. Each *thali* was photographed from four to five viewpoints, and every dish was carefully removed and weighed to the nearest gram on a precision digital scale (±1 g). By leveraging our existing manual segmentation annotations, we then cropped each food region directly from the original photographs—preserving the visual cues of shadows, textures, and apparent volume across all views—before linking them to their measured weights.

The final collection comprised **1394** multi-view images spanning around **267** common *thali* dishes. All weight labels were recorded in raw grams, enabling the models to learn absolute mass predictions. This streamlined structure with paired cutouts, consistent imaging conditions, and precise measurements provides a realistic and focused benchmark for training and evaluating weight regression networks and downstream calorie estimation pipelines.

4.2 Estimating Weights

Our weight estimation pipeline begins by passing each full-plate RGB image through a ResNet-50 backbone [15]. We selected ResNet-50 as it offers a robust and well-established balance between feature extraction power and computational efficiency, providing a strong

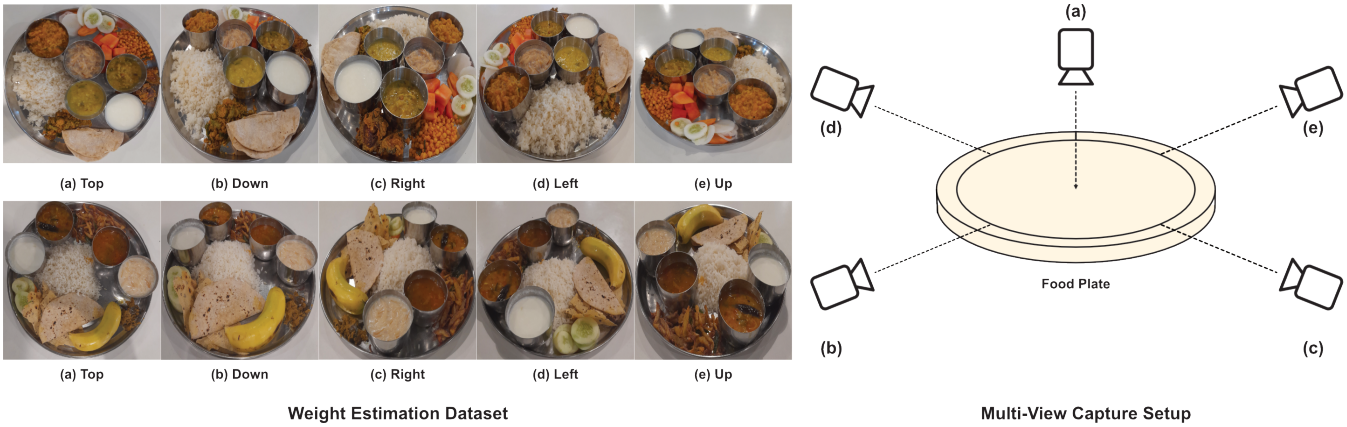

Figure 5: Multi-View Capture Protocol for the Food-Weight Dataset. This figure illustrates the multi-angle imaging setup used to create our novel Food-Weight Dataset. Each Indian *thali* was captured from five distinct viewpoints: Top (a), Down (b), Right (c), Left (d), and Up (e) to capture diverse visual cues for accurate weight estimation of individual dishes. The schematic on the right visually represents these camera positions relative to the food plate.

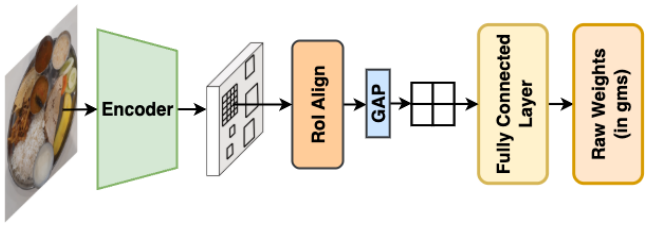

Figure 6: End-to-end pipeline for food weight estimation. Features from the encoder backbone undergo RoI Align and Global Average Pooling before passing through fully connected layers to predict raw dish weights (grams) from segmented *thali* images.

foundation for the regression task without introducing excessive model complexity. We then apply RoI Align [14] with ground-truth masks to obtain fixed-size 7×7 feature maps for each dish.

The model uses an unfrozen ResNet-50 to extract RGB feature maps, from which RoI Align produces fixed 7×7 region features. Each region is globally averaged to a channel vector and fed to a two-layer MLP that outputs class-wise weight predictions. During training, we index each ROI's output by its ground-truth class and apply an L1 loss to that scalar, enabling class-specific weight regression across differing dish scales.

We train end-to-end using Smooth L1 loss optimized with AdamW (learning rate 1×10^{-3}, weight decay 1×10^{-2}) for 100 epochs (batch size 4). Predicted weights convert to calories, protein, carbohydrates, and fat via a per-gram lookup table from Indian recipe databases.

Following weight prediction, we translated each dish's weight into calories and macronutrients using a lookup table. This table, compiled from multiple nutrition sources such as ClearCals, MyNetDiary, Nutritionix, SnapCalorie, FatSecret, Slurrp, and SitaraFoods, provides per-gram values for calories, carbohydrates, proteins, and fats for each of our 41 *thali* dishes. For each RoI, we took the predicted weight (in grams), looked up the corresponding per-gram nutrient densities, and multiplied them to obtain the item's caloric and macronutrient contributions. By decoupling visual weight estimation from nutritional computation, this approach ensures modularity, clear interpretability, easy updates as recipes evolve, and efficient end-to-end calorie estimation from a single photo.

We explore several extensions to the baseline: modality fusion, attention modules, and geometric statistics in our ablation studies.

4.3 Results and Analysis

Category	Type	MAE	MSE	% Error
Weight Estimation	Weight (g)	14.51	501.40	14.62%
Nutrient Estimation†	Calories (kcal)	16.62	803.95	15.76%
	Carbs (g)	2.76	26.29	16.60%
	Protein (g)	0.61	1.22	15.65%
	Fat (g)	0.40	0.58	13.75%

Table 2: Baseline weight and nutrient estimation performance on the test set. †Nutrient predictions are obtained by multiplying predicted weights by per-gram nutrient densities from our lookup table.

The baseline model achieves a mean absolute error (MAE) of 14.679 g, meaning its weight predictions differ from ground truth by an average of about 15 grams per dish. This level of error translates to an average caloric discrepancy of approximately 16.8 kcal, which is within acceptable margins for automated dietary monitoring. The mean squared error (MSE) of 522.074 suggests that while most predictions cluster around the true values, occasional larger deviations—likely arising from irregularly shaped dishes—contribute to higher-squared penalties. The overall percentage error of 20.19% underscores robust performance across both light and heavy dishes.

In nutrient-specific terms, the model's average nutrient losses per dish are 16.803 kcal for calories, 0.607 g for protein, 2.789 g for carbohydrates, and 0.409 g for fat. For a typical *thali* of eight to ten dishes, these errors accumulate to roughly 100 kcal and 3.6 g of macronutrients—still well within acceptable bounds for most dietary-tracking applications. This analysis confirms that the simple RGB-only baseline—using an unfrozen ResNet-50 backbone, 7×7 RoI Align, and a compact MLP head delivers a compelling balance of accuracy and efficiency for real-time nutrition estimation.

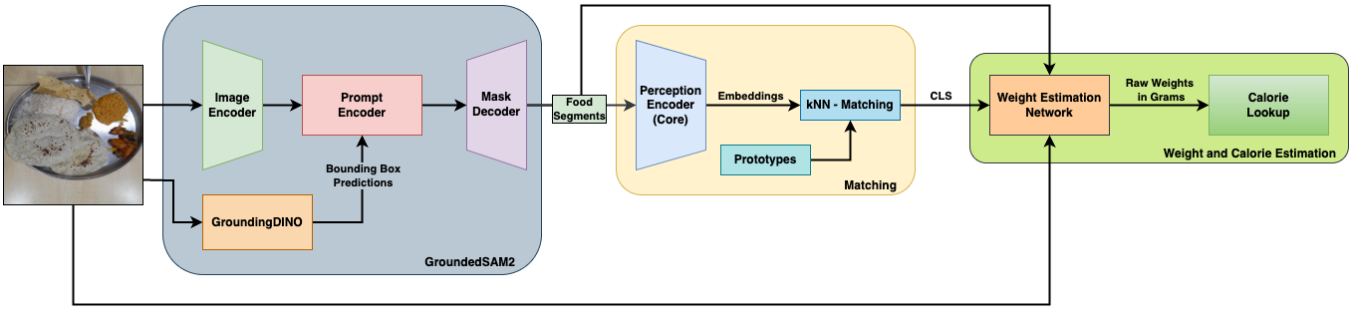

Figure 7: Our Automated Pipeline for Real-Time *thali* Calorie Estimation. This figure outlines the sequential modules of our Food Scanner System: Grounded-SAM-2 (GSAM2) for segmentation, an embedding-based kNN-matching for classification, a dedicated network for weight estimation, and a lookup table for calorie computation, collectively enabling annotation-free, end-to-end nutritional analysis.

Nutrient composition in Indian *thali* components can vary widely due to genetic and environmental factors. In lentils, protein content spans roughly 20-30% on a dry basis, with analyses reporting cultivar-dependent differences across this range [1]. Chickpeas show equally broad variation: protein levels differ significantly among germplasm with reported ranges covering 15.78–31.55% [17] and similar patterns observed in other studies [11]. Mineral content is also influenced by growing conditions, particularly soil micronutrient status—51.2% of Indian soils are zinc-deficient and 19.2% are iron-deficient [9]. Such inherent variability, even before considering preparation or measurement, is substantial enough that per-dish errors in the range of our observed ∼16% for calories lie within expected natural variation.

5 Food Scanner System

The Food Scanner System takes a single image of a *thali* and produces per-dish calorie and macronutrient estimates in real time. It comprises four sequential modules: class-agnostic segmentation, prototype-based classification, weight estimation, and nutrition computation, deployed on an edge device above a university mess serving line. Figure 7 illustrates the end-to-end architecture.

The system is intentionally designed for operational scalability and ease of maintenance. By combining zero-shot segmentation with prototype-based classification, it avoids the need for continuous annotation and retraining cycles. New dishes or menu changes can be accommodated by adding a few representative prototype images, making the system practical for high-throughput cafeterias and other settings where offerings change frequently. This design addresses the deployment challenges of training-heavy supervised pipelines while maintaining reliable nutritional analysis.

5.1 Modular Pipeline

Class-Agnostic Segmentation. We employ GroundedSAM2 (Grounding DINO + SAM2) with a single prompt “food” to obtain binary masks for all edible regions in the image, enabling zero-shot, class-agnostic segmentation without any fine-tuning The robustness of this approach stems from the synergy of the GroundingDINO and SAM2 foundation models, which allows the system to effectively handle natural clutter and complex backgrounds, extending its applicability beyond controlled environments.

Prototype-Based Classification. After segmentation, each mask is cropped and processed as follows:

1. Prototype Gallery Construction.

- Per-Date prototypes: For each menu day, we collect a small set of pre-segmented, manually labeled plates. These day-specific crops are encoded once with our Perception Encoder (the "PE encoder") and stored on disk. The Perception Encoder was selected for its proven ability to generate rich and highly discriminative visual embeddings from image patches. This makes it ideal for our training-free classification module, as the quality of the embeddings is crucial for accurately identifying food items via nearest-neighbor similarity matching against our prototype gallery.
- Global prototypes: We pool all day-wise crops into a single global directory, then at inference filter by the current day’s menu to select relevant embeddings.

2. Patch Sampling and kNN Matching. For each class-agnostic mask, we sample $P = 10$ random patches of size 64×64. Each patch is embedded via the PE encoder and compared using cosine similarity to the stored prototype embeddings. We selected a k-Nearest Neighbors (kNN) approach for its simplicity, interpretability, and training-free scalability. Menu updates only require adding examples to the gallery. Unlike opaque ’black-box’ models, kNN allows for visual inspection of neighbor prototypes, ensuring transparency and easier debugging. We retrieve the top k nearest neighbors ($k \in \{1, 3, 4, 5\}$) per patch.

3. Matching Strategies.

(1) Majority Voting: Each patch casts a vote for its closest prototype; the mask inherits the class with the most votes.
(2) Pooling: For each patch p and class c, let $s_{p,1}^{(c)}, \ldots, s_{p,k}^{(c)}$ be the top-k similarities to prototypes of class c. We compute

$$\text{conf}(c) = \frac{1}{kP} \sum_{p=1}^{P} \sum_{i=1}^{k} s_{p,i}^{(c)},$$

and assign the mask to the class c with highest confidence.

Table 3 summarizes the classification performance (Instance-Level Accuracy, Overall Pixel Accuracy, Class-Agnostic Pixel Accuracy) for both prototype schemes and matching strategies.

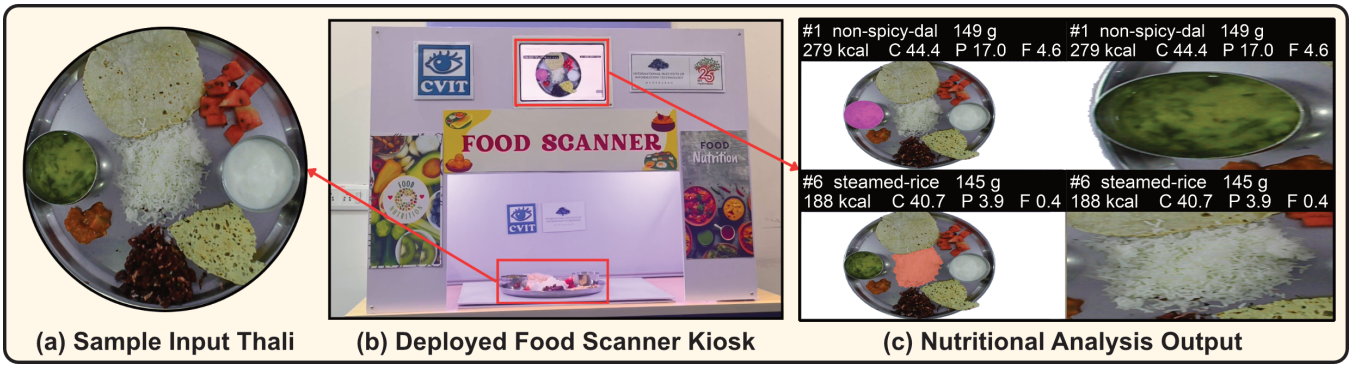

Figure 8: The Food Scanner Kiosk: From Physical Plate to Digital Nutritional Analysis. This figure illustrates the complete, three-stage workflow of the system. First, a user provides a sample multi-item thali as input (a). The thali is then placed in the self-service kiosk for automated image capture (b). Finally, the system processes the image and returns a granular, per-item nutritional breakdown as the final output (c), which includes both full-plate segmentation and detailed close-up views.

Gallery	Strategy	k	ILA (%)	OPA (%)
Per-Date	Majority	1	90.78	92.80
		3	90.92	92.88
		4	90.43	92.64
		5	90.15	92.65
	Pooled	1	90.80	92.79
		3	**91.16**	**93.06**
		4	90.66	92.88
		5	90.29	92.81
Global	Majority	1	84.33	88.01
		3	84.10	87.73
		4	84.10	87.85
		5	83.53	87.34
	Pooled	1	84.42	88.06
		3	**84.51**	**88.09**
		4	84.22	87.97
		5	83.81	87.75

Table 3: Classification performance of Per-Date vs. Global prototypes, under Majority vs. Pooled matching, for various k. CAPA values are omitted here because they remain constant at 96.85% across all configurations. This constancy occurs because the segmentation masks are generated once by the pipeline and remain identical regardless of prototype selection. Best values in bold.

Weight Regression. Each labelled mask is fed via RoI Align into a ResNet-50 backbone followed by a two-layer MLP to predict dish weight in grams. On our Food Weight Dataset, this model achieves a mean absolute error of approximately 14 g.

Nutrition Computation. Predicted weights are converted to calories and macronutrients using a per-gram lookup table; results are summed across all dishes on the plate.

5.2 Validation and Performance

We report Instance-Level Accuracy (ILA), Overall Pixel Accuracy (OPA), and Class-Agnostic Pixel Accuracy (CAPA) introduced in Section 3 (see Equations (1)–(3)). The per-date pooled matching strategy at $k = 3$ yields the best classification performance (ILA = 91.16%, OPA = 93.06%, CAPA = 96.85%), outperforming global prototypes by over 6 pp in ILA and 5 pp in OPA. This suggests that k=3 provides an optimal balance; it is robust enough to mitigate the impact of a single noisy or outlier prototype (unlike k=1) yet constrained enough to avoid introducing ambiguity from less similar prototypes that a larger k might include. Weight regression achieves MAE $\approx$ 14 g.

6 Conclusion

We presented a training-free, scalable Food Scanner system that estimates per-dish weight, calories, and macronutrients from a single *thali* image. By combining zero-shot, class-agnostic segmentation with prototype-based classification and RoI-aligned weight regression, the system achieves up to 91.16% Instance-Level Accuracy, 93.06% Overall Pixel Accuracy, 96.85% Class-Agnostic Pixel Accuracy, and a mean absolute weight error of ~14 g. The use of per-date prototypes further enables the system to handle intra-class variability, for example, the same dish may differ in color or texture on another day due to variations in cooking time, ingredient sourcing, spice proportions, or garnish, as well as inter-class similarity between visually alike dishes such as sambar and dal. These results demonstrate that high accuracy is achievable without training, while maintaining robustness to day-to-day presentation changes, enabling rapid adaptation to new dishes and deployment in diverse food environments. Future work could focus on enhancing the nutrition estimation module. While the current lookup-table approach was a deliberate design choice for modularity and interpretability, it could be replaced by more advanced models that predict nutritional content directly from visual features. Ultimately, by creating both the foundational datasets and a practical deployment pipeline, this work paves the way for a new generation of culturally-aware dietary monitoring tools. Such technology holds immense potential for improving public health outcomes and enabling personalized nutrition at scale, particularly within the diverse culinary landscape of India.

7 Acknowledgements

We thank CDiTH @ IIIT-H for encouraging us to work on this problem.

References

[1] R.S. Bhatty. 1988. Composition and Quality of Lentil (Lens culinaris Medik): A Review. *Canadian Institute of Food Science and Technology Journal* (1988).

[2] Daniel Bolya, Po-Yao Huang, Peize Sun, Jang Hyun Cho, Andrea Madotto, Chen Wei, Tengyu Ma, Jiale Zhi, Jathushan Rajasegaran, Hanoona Rasheed, Junke Wang, Marco Monteiro, Hu Xu, Shiyu Dong, Nikhila Ravi, Daniel Li, Piotr Dollár, and Christoph Feichtenhofer. 2025. Perception Encoder: The best visual embeddings are not at the output of the network.

[3] Lukas Bossard, Matthieu Guillaumin, and Luc Van Gool. 2014. Food-101 – Mining Discriminative Components with Random Forests. In *ECCV*.

[4] Liang-Chieh Chen, Yukun Zhu, George Papandreou, Florian Schroff, and Hartwig Adam. 2018. Encoder-Decoder with Atrous Separable Convolution for Semantic Image Segmentation. In *ECCV*.

[5] Mei Chen, Kapil Dhingra, Wen Wu, Lei Yang, Rahul Sukthankar, and Jie Yang. 2009. PFID: Pittsburgh fast-food image dataset. In *ICIP*.

[6] Yuhao Chen, Jiangpeng He, Chris Czarnecki, Gautham Vinod, Talha Ibn Mahmud, Siddeshwar Raghavan, Jinge Ma, Dayou Mao, Saeejith Nair, Pengcheng Xi, Alexander Wong, Edward Delp, and Fengqing Zhu. 2024. MetaFood3D: Large 3D Food Object Dataset with Nutrition Values. *arXiv preprint arXiv:2409.01966* (2024).

[7] Bowen Cheng, Ishan Misra, Alexander G. Schwing, Alexander Kirillov, and Rohit Girdhar. 2022. Masked-Attention Mask Transformer for Universal Image Segmentation. In *CVPR*.

[8] Sheng-Tzong Cheng, Ya-Jin Lyu, and Ching Teng. 2025. Image-Based Nutritional Advisory System: Employing Multimodal Deep Learning for Food Classification and Nutritional Analysis. *Applied Sciences* (2025).

[9] Salwinder Singh Dhaliwal, Vivek Sharma, Arvind Kumar Shukla, Janpriya Kaur, Vibha Verma, Prabhjot Singh, Harkirat Singh, Shams H Abdel-Hafez, Samy Sayed, Ahmed Gaber, Reham Ali, and Akbar Hossain. 2021. Enrichment of zinc and iron micronutrients in lentil (Lens culinaris Medik.) through biofortification. *Molecules* (2021).

[10] Giovanni Maria Farinella, Dario Allegra, and Filippo Stanco. 2015. A Benchmark Dataset to Study the Representation of Food Images. In *ECCV 2014 Workshops*.

[11] Satvir Kaur Grewal, Kanu Priya Sharma, Rachana D. Bharadwaj, Venkatraman Hegde, Sukhpreet Kaur Sidhu, Sarvjeet Singh, Pradeep Kumar Jain, Souliha Rasool, Dev Kumar Arya, Pawan Kumar Agrawal, and et al. 2022. Characterization of chickpea cultivars and trait specific germplasm for grain protein content and amino acids composition and identification of potential donors for genetic improvement of its nutritional quality. *Plant Genetic Resources: Characterization and Utilization* (2022).

[12] Meng-Hao Guo, Cheng-Ze Lu, Qibin Hou, Zheng-Ning Liu, Ming-Ming Cheng, and Shi min Hu. 2022. SegNeXt: Rethinking Convolutional Attention Design for Semantic Segmentation. In *NIPS*.

[13] Rakib Ul Haque, Razib Hayat Khan, A S M Shihavuddin, M M Mahbubul Syeed, and Mohammad Faisal Uddin. 2022. Lightweight and parameter-optimized real-time food calorie estimation from images using CNN-based approach. *Applied Sciences (Basel)* (2022).

[14] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. 2017. Mask R-CNN. In *ICCV*.

[15] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep Residual Learning for Image Recognition. In *CVPR*.

[16] Hajime Hoashi, Taichi Joutou, and Keiji Yanai. 2010. Image Recognition of 85 Food Categories by Feature Fusion. In *2010 IEEE International Symposium on Multimedia*.

[17] Uday Chand Jha, Harsh Nayyar, Mahender Thudi, Radha Beena, P. V. Vara Prasad, and Kadambot H. M. Siddique. 2024. Unlocking the nutritional potential of chickpea: strategies for biofortification and enhanced multinutrient quality. *Frontiers in Plant Science* (2024).

[18] Jing jing Chen and Chong wah Ngo. 2016. Deep-based Ingredient Recognition for Cooking Recipe Retrieval. *ACM Multimedia* (2016).

[19] Yoshiyuki Kawano and Keiji Yanai. 2014. FoodCam-256: A Large-scale Real-time Mobile Food Recognition System employing High-Dimensional Features and Compression of Classifier Weights. In *Proceedings of the 22nd ACM International Conference on Multimedia*. Association for Computing Machinery.

[20] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C. Berg, Wan-Yen Lo, Piotr Dollar, and Ross Girshick. 2023. Segment Anything. In *ICCV*.

[21] Fotios Konstantakopoulos, Eleni Georga, and Dimitrios Fotiadis. 2023. A Review of Image-Based Food Recognition and Volume Estimation Artificial Intelligence Systems. *IEEE reviews in biomedical engineering* (2023).

[22] Jonathan Long, Evan Shelhamer, and Trevor Darrell. 2015. Fully Convolutional Networks for Semantic Segmentation. In *CVPR*.

[23] Ya Lu, Thomai Stathopoulou, Maria F Vasiloglou, Lillian F Pinault, Colleen Kiley, Elias K Spanakis, and Stavroula Mougiakakou. 2020. GoFOODTM: An artificial intelligence system for dietary assessment. *Sensors (Basel)* (2020).

[24] Timo Lüddecke and Alexander Ecker. 2022. Image Segmentation Using Text and Image Prompts. In *CVPR*.

[25] Weiqing Min, Shuqiang Jiang, Linhu Liu, Yong Rui, and Ramesh Jain. 2019. A Survey on Food Computing. *ACM Comput. Surv.* (2019).

[26] Austin Myers, Nick Johnston, Vivek Rathod, Anoop Korattikara, Alex Gorban, Nathan Silberman, Sergio Guadarrama, George Papandreou, Jonathan Huang, and Kevin Murphy. 2015. Im2Calories: Towards an Automated Mobile Vision Food Diary. In *ICCV*.

[27] Nikhila Ravi, Valentin Gabeur, Yuan-Ting Hu, Ronghang Hu, Chaitanya Ryali, Tengyu Ma, Haitham Khedr, Roman Rädle, Chloe Rolland, Laura Gustafson, Eric Mintun, Junting Pan, Kalyan Vasudev Alwala, Nicolas Carion, Chao-Yuan Wu, Ross Girshick, Piotr Dollar, and Christoph Feichtenhofer. 2025. SAM 2: Segment Anything in Images and Videos. In *The Thirteenth International Conference on Learning Representations*.

[28] Tianhe Ren, Shilong Liu, Ailing Zeng, Jing Lin, Kunchang Li, He Cao, Jiayu Chen, Xinyu Huang, Yukang Chen, Feng Yan, Zhaoyang Zeng, Hao Zhang, Feng Li, Jie Yang, Hongyang Li, Qing Jiang, and Lei Zhang. 2024. Grounded SAM: Assembling Open-World Models for Diverse Visual Tasks.

[29] Shayan Rokhva, Babak Teimourpour, and Amir Hossein Soltani. 2024. Computer Vision in the Food Industry: Accurate, Real-Time, and Automatic Food Recognition with Pretrained MobileNetV2. *Food and Humanity* (2024).

[30] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. 2015. U-net: Convolutional networks for biomedical image segmentation. In *MICCAI*. Springer.

[31] Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. 2018. MobileNetV2: Inverted Residuals and Linear Bottlenecks. In *CVPR*.

[32] Ashutosh Singla, Lin Yuan, and Touradj Ebrahimi. 2016. Food/Non-food Image Classification and Food Categorization using Pre-Trained GoogLeNet Model. In *Proceedings of the 2nd International Workshop on Multimedia Assisted Dietary Management*. Association for Computing Machinery.

[33] Ghalib Ahmed Tahir and Chu Kiong Loo. 2021. A Comprehensive Survey of Image-Based Food Recognition and Volume Estimation Methods for Dietary Assessment. *Healthcare (Basel)* (2021).

[34] Quin Thames, Arjun Karpur, Wade Norris, Fangting Xia, Liviu Panait, Tobias Weyand, and Jack Sim. 2021. Nutrition5k: Towards Automatic Nutritional Understanding of Generic Food. In *CVPR*.

[35] Stephanie Van Asbroeck and Christophe Matthys. 2020. Use of different food image recognition platforms in dietary assessment: Comparison study. *JMIR Form. Res.* (2020).

[36] Hui Wang, Haixia Tian, Ronghui Ju, Liyan Ma, Ling Yang, Jingyao Chen, and Feng Liu. 2024. Nutritional composition analysis in food images: an innovative Swin Transformer approach. *Frontiers in Nutrition* (2024).

[37] Xiongwei Wu, Xin Fu, Ying Liu, Ee-Peng Lim, Steven CH Hoi, and Qianru Sun. 2021. A Large-Scale Benchmark for Food Image Segmentation. In *Proceedings of ACM International Conference on Multimedia*.

[38] Enze Xie, Wenhai Wang, Zhiding Yu, Anima Anandkumar, Jose M Alvarez, and Ping Luo. 2021. SegFormer: Simple and Efficient Design for Semantic Segmentation with Transformers. In *NIPS*.

[39] Qinqiu Zhang, Chengyuan He, Wen Qin, Decai Liu, Jun Yin, Zhiwen Long, Huimin He, Ho Ching Sun, and Huilin Xu. 2022. Eliminate the hardware: Mobile terminals-oriented food recognition and weight estimation system. *Frontiers in Nutrition* (2022).

[40] Yaping Zhao, Ping Zhu, Yizhang Jiang, and Kaijian Xia. 2024. Visual nutrition analysis: leveraging segmentation and regression for food nutrient estimation. *Frontiers in Nutrition* (2024).

[41] Lei Zhou, Chu Zhang, Fei Liu, Zhengjun Qiu, and Yong He. 2019. Application of Deep Learning in Food: A Review. *Compr Rev Food Sci Food Saf* (2019).