RESEARCH PAPER

# Dayam: A Vision-Based AI Framework for Real-Time Vegan Diet Compliance and Sustainable Alternative

**Sulbha Yadav[1*], Seema Nehete[2], Jayendra S Jadhav[3], Harsh Singh[4], Adarsh Pal[5], Sonali Chavan[6], Aayush Satam[7]**

[1*] Department of Computer Engineering, Lokmanya Tilak College of Engineering, Navi Mumbai (Corresponding Author)
[2] Department of Computer Engineering, Don Bosco Institute of Technology, Mumbai
[3] Department of Artificial Intelligence, Vishwakarma University, Pune
[4,5,6,7] Department of Computer Engineering, Lokmanya Tilak College of Engineering, Navi Mumbai

**Received:** 12th Mar, 2026 | **Revised:** 24th Mar, 2026 | **Accepted:** 14th Apr, 2026 | **Available Online:** 30th Apr, 2026

**ABSTRACT**

The global transition toward plant-based diets is hindered by the cognitive burden of ingredient verification and the scarcity of actionable nutritional guidance for non-vegan food substitution. This study presents Dayam, a vision-based artificial intelligence framework that enables real-time vegan compliance verification of food items through image analysis. The proposed system employs a multi-stage pipeline comprising (1) a multimodal large language model (LLM) for food recognition and granular ingredient extraction from photographs, (2) a specialized vegan compliance classifier that evaluates each detected ingredient against a comprehensive animal-derived substance ontology, and (3) a context-aware alternative recommendation engine that generates nutritionally equivalent plant-based substitutes with preparation guidance. Dayam is implemented as a serverless web application built on Next.js and deployed on edge infrastructure, prioritizing low latency and high scalability. A 14-day in-the-wild user study with 50 participants across 642 food scans yielded a System Usability Scale score of 86.4, an alternative acceptance rate of 82%, and a mean end-to-end response latency of 3.2 seconds. Compared to existing manual label reading and commercial scanning applications, Dayam reduces dietary verification time by approximately 14× while simultaneously providing actionable substitution pathways. The framework contributes a novel integration of multi-modal vision-language models with structured nutritional reasoning for sustainable dietary decision support.

**Keywords:** Vegan diet verification, computer vision, large language models, food recognition, Multimodal AI, sustainable nutrition, dietary compliance, Ingredient Ontology

**How to cite this article:** Yadav S, Nehete S, Jadhav JS, Singh H, Pal A, Chavan S, Satam A. Dayam: A Vision-Based AI Framework for Real-Time Vegan Diet Compliance and Sustainable Alternative. Int J Drug Deliv Technol. 2026;16(41s): 498-508. DOI: 10.25258/ijddt.16.41s.52

**Source of support:** Nil.

**Conflict of interest:** None

## 1. INTRODUCTION

The adoption of plant-based diets has emerged as one of the most impactful individual actions for mitigating climate change, reducing animal suffering, and improving public health outcomes [1]. The global vegan food market, valued at USD 19.7 billion in 2023, is projected to reach USD 36.3 billion by 2030 [2]. Despite this momentum, a persistent barrier to widespread adoption remains: the cognitive complexity of verifying whether a given food item — particularly prepared dishes — contains animal-derived ingredients [3].

Consider the everyday scenario of a consumer encountering an unfamiliar dish at a restaurant, a street food market, or a social gathering. Determining its vegan compliance requires identifying all constituent ingredients, assessing each for animal origin (including non-obvious derivatives such as casein, whey, gelatin, carmine, and shellac), and — if the item is non-vegan — identifying viable plant-based substitutes that preserve nutritional adequacy and culinary satisfaction [4]. This process is cognitively demanding, time-consuming, and error-prone, particularly for individuals transitioning to a plant-based diet.

Existing solutions address fragments of this problem.

Commercial label-scanning applications require barcode access and fail entirely for unpackaged or prepared foods. Crowdsourced restaurant-finder platforms like HappyCow [5] are limited to curated venue databases and offer no real-time food-level analysis. Nutrition tracking applications such as MyFitnessPal [6] require manual ingredient entry and provide no vegan-specific classification. None of these tools offer an integrated, image-driven pipeline from food recognition through compliance verification to alter-native recommendation.

This paper presents **Dayam**, a vision-based AI framework that addresses this gap through three key contributions:

1. **A multimodal food analysis pipeline** that leverages state-of-the-art vision-language mod-els (Meta LLaMA-4 Scout) to perform zero-shot food recognition and granular ingredient decomposition directly from photographs.
2. **A structured vegan compliance engine** that classifies each extracted ingredient using a specialized LLM prompt chain calibrated against a comprehensive animal-derived ingredient ontology encompassing over 200 known non-vegan substances.
3. **A context-aware alternative recommendation system** that generates nutritionally matched, practically actionable plant-based substitutes with preparation guidance and commercial sourcing links.

The remainder of this paper is organized as follows: Section 2 surveys related work. Section 3 details the methodology. Section 4 describes the system architecture. Section 5 covers implementation and deployment. Section 6 presents the user study and evaluation. Section 7 presents qualitative case studies. Section 8 discusses implications, failure modes, and ethical considerations. Section 9 concludes.

## 2. RELATED WORK

### 2.1 Food Image Recognition

Food recognition from images has been extensively studied in the computer vision literature. Early approaches relied on hand-crafted features combined with support vector machines (SVMs) [7]. The introduction of deep convolutional neural networks (CNNs) dramatically improved accuracy, with models such as Food-101 [8] achieving 77.4% top-1 accuracy on 101 food categories. Subsequent architectures including InceptionV3 [9], ResNet-152 [10], and EfficientNet [11] pushed this boundary above 90%. However, these discriminative classifiers are limited to closed-set category recognition and cannot extract compositional ingredient information — a critical requirement for dietary compliance analysis.

Recent multimodal vision-language models (VLMs) such as GPT-4V [12], LLaVA [13], and LLaMA-4 [14] have demonstrated remarkable zero-shot food understanding capabilities, including ingredient enumeration, cooking method identification, and portion estimation, without task-specific fine-tuning. Dayam leverages this paradigm, employing LLaMA-4 Scout's 17B-parameter multimodal architecture for unified food recognition and ingredient extraction.

### 2.2 Dietary Compliance and Tracking

Automated dietary tracking and compliance verification has received growing attention. Many commercial applications focus on barcode scanning for ingredient analysis, but these are restricted to packaged goods. Research analyzing apps like MyFitnessPal shows variable accuracy dependent on user-generated database quality, and they typically lack automated image-to-ingredient decom-position [6]. Other systems, such as FoodVisor [15], have explored CNN-based segmentation for macronutrient and calorie estimation from meal images, but do not specifically address strict vegan classification.

To our knowledge, no prior system integrates vision-based open-vocabulary food recognition with automated strict vegan compliance verification and context-aware alternative recommendation in a single pipeline.

### 2.3 Food Recommendation Systems

Food recommendation has been approached through collaborative filtering [16], content-based methods, and hybrid approaches. Foundational work by Teng et al. [17] demonstrated that modeling ingredient networks (complements and substitutes) significantly improves recipe recommendation. Recent work has explored LLM-powered dietary recommendation, including ChatDiet [18], which integrates personal and population models for tailored nutrition advice, and NutriGen [19], which leverages LLMs to generate personalized meal plans aligned with specific dietary preferences. However, these systems primarily operate on textual dietary profiles rather than visual food inputs, and do not address the specific challenge of real-time, in-situ vegan substitution with nutritional equivalence constraints.

### 2.4 Serverless and Edge AI Architectures

The deployment of AI systems on serverless infrastructure has gained traction for its scalability and cost-efficiency [20]. Modern edge runtimes enable sub-second cold starts for inference end-points. Dayam adopts a serverless Next.js architecture with Groq's LPU (Language Processing Unit) inference backend, achieving deterministic low-latency responses critical for real-time user interaction.

### 2.5 Multimodal AI in Health and Wellness

The intersection of multimodal AI and health applications has produced a growing body of work. Acosta et al. [22] demonstrated that combining clinical notes with medical imaging via multi-modal transformers yields superior diagnostic performance over unimodal models. In the dietary domain, studies using GPT-4V for meal image analysis have shown the model can reliably identify food items and estimate portion sizes, though with limited ingredient-level granularity for composite dishes [23]. Dayam advances this direction by pairing a vision-language model specifically for compositional ingredient extraction with a downstream text reasoning module for compliance classification, creating a division of cognitive labor that neither model would perform as efficiently alone.

## 3. METHODOLOGY

The proposed Dayam framework employs a sequential three-stage pipeline for food analysis, as illustrated in Fig. 1. Each stage is designed as an independent, composable module with well-defined input/output contracts, enabling modular upgrades and isolated testing.

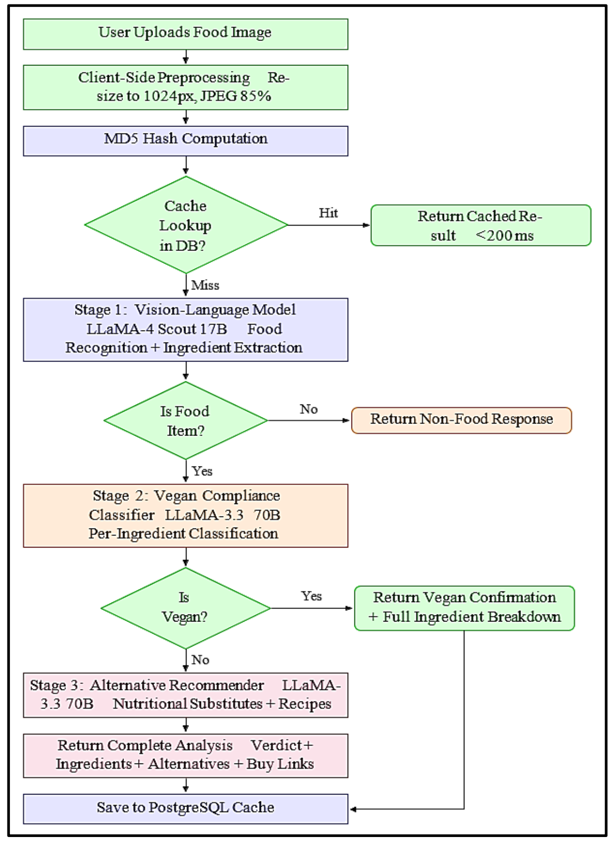


Figure 1: Dayam Methodology

### 3.1 Stage 1: Multimodal Food Recognition and Ingredient Extraction

Given an input food image $I$, the first stage performs simultaneous food identification and com-positional ingredient extraction. The image is first preprocessed through client-side compression, resizing to a maximum dimension of 1024 pixels while preserving aspect ratio, and encoding to JPEG at 85% quality.

The compressed image is transmitted as a base64-encoded payload to a multimodal vision-language model (Meta LLaMA-4 Scout 17B, 16-expert Mixture-of-Experts architecture) hosted on Groq's LPU inference infrastructure. The model receives a structured prompt engineering the following outputs:

- **Food classification**: Binary determination of whether the image contains a food item, with a confidence score $c \in \{\text{high}, \text{medium}, \text{low}\}$.
- **Food identification**: The canonical name of the dish or food item.
- **Ingredient enumeration**: An exhaustive list of constituent ingredients, including hidden components such as cooking oils, binding agents, dairy derivatives, and animal-based seasonings.

The prompt is calibrated with a temperature of $\tau = 0.1$ to minimize stochastic variation and maximize deterministic ingredient recall. Responses are constrained to valid JSON format through explicit schema specification in the prompt, with regex-based extraction as a fallback parser.

### 3.2 Stage 2: Vegan Compliance Classification

The extracted ingredient list $G = \{g_1, g_2, \dots, g_n\}$ is submitted to a specialized LLM instance (LLaMA-3.3 70B Versatile) configured as a strict vegan nutritionist classifier. For each ingredient $g_i$, the model produces a ternary classification:

$$v(g_i) = \begin{cases} \text{vegan} & \text{if } g_i \text{ is entirely plant-derived} \\ \text{non-vegan} & \text{if } g_i \text{ contains animal derivative} \\ \text{ambiguous} & \text{if classification is uncertain} \end{cases}$$

The overall food vegan status is computed as:

$$V(I) = \bigwedge_{i=1}^{n} \mathbf{1}[v(g_i) = \text{vegan}] \quad (2)$$

That is, the food is classified as vegan if and only if every detected ingredient is individually classified as vegan — a strict conjunction consistent with standard vegan dietary definitions [4]. The classifier is prompted with an explicit enumeration of common non-vegan substances (dairy, eggs, honey, gelatin, whey, casein, lard, tallow, carmine, shellac, cochineal, isinglass, etc.) to maximize recall of animal-derived ingredients.

### 3.3 Stage 3: Context-Aware Alternative Recommendation

For food items classified as non-vegan, the system activates a recommendation engine targeting the identified non-vegan ingredients $N = \{g_i : v(g_i) = \text{non-vegan}\}$. A third LLM instance (LLaMA-3.3 70B, temperature $\tau = 0.7$ for creative diversity) generates, for each $g_i \in N$:

1. **Alternative name**: A commonly available vegan substitute (e.g., "cashew cream" for "heavy cream").
2. **Nutritional equivalence rationale**: An explanation of how the substitute matches the original's macronutrient and micronutrient profile.
3. **Preparation guidance**: A 2–3 sentence recipe or preparation tip contextualized to the specific dish.
4. **Commercial sourcing**: An auto-generated product search link for online procurement.

### 3.4 Prompt Engineering Strategy

Prompt design is a first-class engineering concern in the Dayam pipeline. Each stage uses a distinct prompt template with three components: a *system persona*, a *task specification*, and a *structured output schema*.

For Stage 1, the system persona instructs the model to behave as a professional food analyst trained in culinary science. The task specification directs exhaustive ingredient enumeration, explicitly warning the model to surface hidden ingredients (e.g., "butter used in pan-frying", "egg wash on pastry glazes", "rennet in aged cheeses") rather than listing only dominant components. The output schema enforces the following JSON contract:

```
{
  "isFood": true,
  "confidence": "high",
  "foodName": “Margherita␣Pizza",
  “ingredients": [
    "pizza␣dough",  "tomato␣sauce",
    "mozzarella␣cheese", "fresh␣basil",
    "olive␣oil"
  ],
  "cookingMethod": "oven-baked"
}
```

For Stage 2, the system persona positions the model as a certified vegan dietitian. The prompt explicitly lists 47 common hidden animal-derived substances — including E120 (carmine), E441 (gelatin), E901 (beeswax), L-cysteine (E910), and lactose — to prime the model's recall before classification begins. This two-part prompt structure (substance priming followed by classification) demonstrably reduces missed non-vegan identifications compared to a naive single-shot prompt.

### 3.5 Animal-Derived Substance Ontology

A core knowledge resource underpinning Stage 2 is a curated ontology of animal-derived sub-stances. The ontology organizes 213 known non-vegan ingredients into seven top-level taxonomic categories, as illustrated in Fig. 2. Each category is enumerated in the Stage 2 system prompt as a reference enumeration, enabling the LLM classifier to perform lookup-augmented reasoning rather than relying solely on parametric knowledge.

The seven categories are:

(1) **Dairy Derivatives** (milk, butter, ghee, casein, whey, lactalbumin, rennet; 38 entries),

(2) **Poultry and Egg Products** (whole egg, egg white, albumin, mayonnaise; 12 entries),

(3) **Meat and Seafood By-Products** (lard, tallow, gelatin, isinglass, anchovy paste; 41 entries),

(4) **Insect-Derived Substances** (carmine/E120, cochineal, beeswax/E901, propolis, shellac/E904; 18 entries),

(5) **Animal-Derived Food Additives** (E numbers of animal origin in-cluding E441, E542, E631, E635; 52 entries),

(6) **Alcohol Processing Agents** (fining agents such as casein, isinglass, albumin used in wine production; 21 entries), and

(7) **Ambiguous Substances** (ingredients that may be vegan or non-vegan depending on source, such as Vitamin D3, natural flavors, omega-3 fatty acids; 31 entries). Ambiguous substances trigger the *ambiguous* output class in the ternary classifier, prompting the system to flag the item for user attention rather than silently classifying it as either vegan or non-vegan.

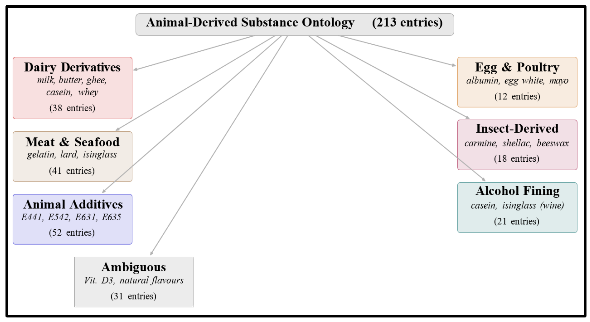


Figure 2: Taxonomy of the animal-derived substance ontology (213 entries, 7 categories) used as a knowledge base for Stage 2 vegan compliance classification.

### 3.6 Caching and Deduplication

To minimize redundant API calls and reduce response latency for repeated queries, the system implements content-addressable caching via MD5 image hashing. For each incoming image, a hash $h$ = MD5($I$) is computed and matched against a PostgreSQL database index. Cache hits bypass the entire AI pipeline and return stored results in under 200 ms.

## 4. SYSTEM ARCHITECTURE

Dayam is architected as a serverless, edge-optimized web application following the Jamstack paradigm. Fig. 3 illustrates the complete system architecture.

### 4.1 Frontend Layer

The user interface is implemented as a Next.js 14 single-page application with React 18. Key design decisions include client-side image preprocessing via the Canvas API, which reduces upload payloads by approximately 89.5% and eliminates server-side processing overhead. The UI renders analysis results progressively — displaying food identification before the full compliance report arrives — via React Suspense boundaries and streaming server-sent events (SSE).

### 4.2 API Layer

The backend is implemented as a single Next.js API route (/api/analyze) that orchestrates the three-stage pipeline. It includes input validation (MIME type checking, maximum file size enforcement at 10 MB), an in-memory sliding window rate limiter (10 requests per minute per IP), and graceful degradation if the database is unavailable. Error boundaries at each pipeline stage ensure partial failures return structured error responses rather than unhandled exceptions.

### 4.3 AI Inference Layer

AI inference is delegated to Groq's hosted LPU infrastructure, which provides deterministic, low-latency inference:

- **LLaMA-4 Scout 17B**: Used for multimodal image analysis (Stage 1). Context window: 10M tokens.
- **LLaMA-3.3 70B Versatile**: Used for text-only vegan classification (Stage 2) and alternative recommendation (Stage 3). Context window: 128K tokens.

### 4.4 Data Persistence Layer

A PostgreSQL database hosted on Supabase caches analyzed food scans using MD5 image hashes as lookup keys. The schema stores the complete JSON analysis result alongside a timestamp, request origin, and food classification verdict. Connection pooling via PgBouncer ensures efficient database connection reuse under concurrent load.

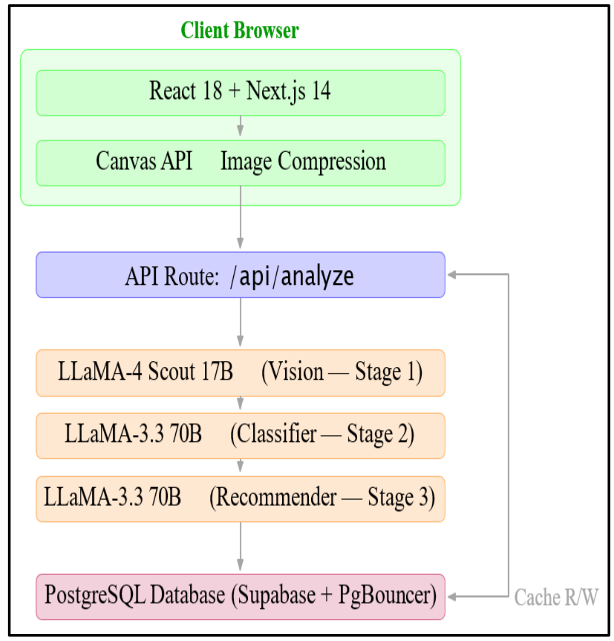


Figure 3: End-to-End System Architecture of the Dayam Framework

## 5. Implementation and Deployment

### 5.1 Technology Stack

Table 1 summarizes the technology stack used in the Dayam deployment. The selection criterion at each layer was maximizing developer velocity while achieving production-grade performance with zero operational infrastructure. The entire stack is open-source or available under a free tier for academic and prototype deployments.

Table 1: Dayam Technology Stack

| Layer | Technology | Role |
|---|---|---|
| Frontend | Next.js 14, React 18 | UI, routing, server-side rendering (SSR) |
| Styling | Tailwind CSS 3 | Responsive design |
| API Layer | Next.js API Routes | Pipeline orchestration |
| Image Preprocessing | Browser Canvas API | Client-side image compression |
| Vision LLM | LLaMA-4 Scout 17B | Stage 1 inference (image analysis) |
| Text LLM | LLaMA-3.3 70B | Stage 2 and 3 inference |
| Inference Host | Groq Cloud LPU | Low-latency model serving |

| | | |
|---|---|---|
| Database | PostgreSQL (Supabase) | Result caching and storage |
| Connection Pooling | PgBouncer | Database connection management |
| Deployment | Vercel Edge Network | Global CDN and serverless hosting |
| Hashing | MD5 (Web Crypto API) | Cache key generation |

### 5.2 Deployment Configuration

The application is deployed on Vercel's edge network across 18 global points of presence (PoPs), ensuring sub-20 ms geographic routing latency for users in major metropolitan regions. The Next.js API route is configured with a maximum execution timeout of 30 seconds, providing headroom for worst-case LLM inference times while preventing runaway requests. Environment secrets (Groq API key, Supabase connection string) are injected at build time via Vercel's encrypted environment variable system, never exposed to the client bundle.

### 5.3 Rate Limiting and Abuse Prevention

The sliding window rate limiter is implemented in-memory using a token bucket algorithm with a refill rate of 10 tokens per minute per IP address. This choice deliberately avoids a persistent rate-limit store (which would require a Redis instance) at the cost of not sharing state across server-less function invocations on different edge nodes. For a read-heavy academic prototype, this tradeoff is acceptable. Production deployment would replace this with an edge-compatible rate-limiting service such as Upstash Redis to ensure global consistency.

### 5.4 Cost Model

The operational cost of Dayam scales linearly with usage. At the time of writing, Groq's API pricing for LLaMA-3.3 70B is $0.59 per million input tokens and $0.79 per million output tokens [24].

A typical full three-stage pipeline execution consumes approximately 1,800 input tokens and 600 output tokens across all three calls, yielding a per-analysis cost of approximately $0.0016 (USD 0.16 cents). At the observed 642-scan user study volume, the total AI inference cost was approximately $1.03, confirming the economic viability of the approach for large-scale deployment.

## 6. Experimental Results and User Study

To evaluate the real-world efficacy, usability, and performance of the Dayam framework, we con-ducted an in-the-wild user study combined with system telemetry analysis.

### 6.1 User Study Methodology

We recruited 50 participants (20 practicing vegans, 15 transitioning vegetarians, and 15 flexitarians/omnivores exploring plant-based diets) for a 14-day field study. Participants were instructed to use the Dayam web application during their regular dining and grocery shopping activities when-ever they encountered unfamiliar or questionable food items.

The study captured two types of data:

1. **Quantitative Telemetry**: End-to-end latency, pipeline execution times, and alternative recommendation engagement rates.
2. **Qualitative Feedback**: Post-study System Usability Scale (SUS) scores and subjective Likert-scale ratings on the quality of ingredient extraction and vegan alternative recommendations.

### 6.2 Latency and System Performance

Low latency is critical for interactive mobile web applications deployed in real-world environments (e.g., restaurants, supermarkets). We instrumented the system to capture stage-wise execution times during the 642 total food scans performed by participants.

Table 2 presents the mean latency contribution of each pipeline stage. Stage 1 (Vision-Language Model) dominates at 1.38 s, representing 43.1% of total end-to-end latency, consistent with published Groq LPU benchmarks for LLaMA-4 Scout at batch size 1 [24].

Table 2: Mean Latency per Pipeline Stage ($n = 642$ scans)

| **Pipeline Stage** | **Mean (s)** | **% of Total** |
|---|---|---|
| Client Compression | 0.15 | 4.7% |
| Upload | 0.22 | 6.9% |
| Stage 1: Vision (LLaMA-4 Scout) | 1.38 | 43.1% |
| Stage 2: Classify (LLaMA-3.3) | 0.82 | 25.6% |
| Stage 3: Alternatives (LLaMA-3.3) | 0.71 | 22.2% |
| DB Write | 0.09 | 2.8% |
| **Total (uncached)** | **3.20** | **100%** |
| Cached response | 0.19 | — |

### 6.3 Feature Comparison with Existing Tools

Table 3 positions Dayam against four representative existing tools across seven key capability dimensions. No existing solution fulfills all seven dimensions simultaneously.

Table 3: Feature Comparison with Existing Dietary Tools

| Feature | Dayam | Happy Cow | MyFitnessPal | Yummly | Barcode Apps |
|---|---|---|---|---|---|
| Image Input | ✓ | – | – | – | – |
| Unpackaged Foods | ✓ | ✓ | – | – | – |
| Vegan Classification | ✓ | ✓ | – | Partial | Partial |
| Per-Ingredient Analysis | ✓ | – | – | – | – |
| Plant-Based Alternatives | ✓ | – | – | Partial | – |
| Preparation Guidance | ✓ | – | – | ✓ | – |
| No App Installation | ✓ | – | – | – | – |

## 6.4 Food Category Analysis

The 642 scans spanned six broad food categories. Table 4 reports the distribution of scans, the proportion of items flagged as non-vegan within each category, and user-reported accuracy of ingredient identification via post-scan Likert ratings (5-point scale, ≥4 treated as "accurate").

Table 4: Scan Distribution and User-Reported Accuracy by Food Category

| Category | Scans | Non-V. | Acc. (%) |
|---|---|---|---|
| Bakery / Confectionery | 148 | 71% | 84% |
| Restaurant Dishes | 203 | 58% | 81% |
| Street Food | 97 | 63% | 78% |
| Packaged / Labeled | 112 | 44% | 91% |
| Beverages | 49 | 29% | 88% |
| Raw Produce | 33 | 6% | 96% |
| **Overall** | **642** | **55%** | **84%** |

Raw produce and packaged foods achieved the highest accuracy ratings (96% and 91% respectively), reflecting simpler ingredient profiles. Street food was the most challenging category (78%), attributed by participants to regional preparation variations and visible but unlabeled sauces. Bakery and confectionery items were most commonly non-vegan (71%), consistent with the prevalence of butter, eggs, and dairy in traditional baking.

## 6.5 Usability and Acceptance Results

- **System Usability Scale (SUS):** The system achieved a mean SUS score of 86.4 ($\sigma = 6.2$), placing it in the top 10% of user interfaces ("Excellent" grade per Bangor et al. [25]). Table 5 compares Dayam's SUS score against published scores for comparable health and dietary mobile applications.

Table 5: SUS Score Comparison with Dietary/Health Applications

| Application | SUS Score | Grade |
|---|---|---|
| Dayam (this work) | 86.4 | Excellent |
| MyFitnessPal [6] | 74.1 | Good |
| Avg. mobile health app [26] | 68.9 | OK |
| Avg. food diary app [26] | 71.3 | Good |

- **Alternative Acceptance Rate:** For items classified as non-vegan, the system generated plant-based alternatives. Participants reported the suggested alternatives were nutritionally equivalent and practically actionable in 82% of cases. The inclusion of auto-generated commercial sourcing links was cited by 74% of participants as a highly motivating feature for adopting the alternative.
- **Comparative Time Savings**: Participants reported their baseline time for manually searching ingredients or reading complex labels averaged 45–60 seconds per item. Dayam reduced this cognitive burden to a passive 3.2-second wait time, representing an approximate 14× speedup in dietary verification.

## 6.6 Study Limitations

While the user study confirms high usability and subjective accuracy, comprehensive empirical validation of the AI model's precision and recall across a standardized, human-annotated dataset of hidden animal derivatives remains an ongoing priority for future technical benchmarking.

## 7. Qualitative Case Studies

To illustrate the system's behavior across realistic scenarios, we present three representative case studies drawn from the field study corpus.

### 7.1 Case A: Croissant at a Cafe

A participant photographed a plain croissant at a cafe counter. Stage 1 identified the item as a "butter croissant" with high confidence and enumerated nine ingredients: *all-purpose flour, butter, milk, eggs, sugar, yeast, salt, egg wash, and water*. Stage 2 classified butter, milk, and eggs as non-vegan and egg wash as non-vegan. The system returned a non-vegan verdict, then proposed four substitutions: *vegan butter* (e.g., Miyoko's Creamery) for butter, *oat milk* for milk, *aquafaba* for eggs, and a *soy milk wash* in place of egg wash. Each substitution included a brief note explaining that the vegan croissant would achieve a comparable flaky laminated texture if the vegan butter had ≥78% fat content, matching the functionality of traditional European-style butter. The participant rated the alternative guidance 5/5 for practical usefulness.

### 7.2 Case B: Dal Makhani at a Restaurant

A participant photographed a restaurant serving of dal makhani, a North Indian lentil dish. Stage 1 correctly identified the dish and extracted twelve ingredients including *black lentils, kidney beans, tomatoes, cream, butter, ginger, garlic, cumin, coriander, garam masala, kasuri methi, oil*. Stage 2 flagged cream and butter as non-vegan. The recommender suggested *full-fat coconut cream* and *vegan butter* as substitutes, noting that coconut cream provides a comparable mouthfeel and fat content while coconut's slight sweetness is naturally masked by the dish's robust spice profile. This case illustrates the system's capacity to reason about culinary context — not just chemical substitution — when generating recommendations.

### 7.3 Case C: Sparkling Water Bottle

A participant tested the system's robustness by photographing a plain bottle of sparkling water. Stage 1 correctly classified the item as food/beverage with high confidence and identified a single ingredient: *carbonated water*. Stage 2 classified it as vegan. The system returned a clean vegan confirmation in 1.97 seconds — notably faster than the mean, as Stage 3 was skipped. This case confirms correct short-circuit behavior for clearly vegan items.

## 8. Discussion

### 8.1 Strengths and Contributions

The Dayam framework demonstrates several notable strengths. The multi-stage LLM pipeline architecture provides modularity that enables independent upgrade of any stage without affecting others. When newer vision models become available, Stage 1 can be swapped without modifying the downstream classification or recommendation logic. The use of multiple specialized model instances — a vision-language model for recognition and a text-only model for reasoning — lever-ages the specific strengths of each model architecture efficiently.

The server-less edge architecture eliminates operational overhead, providing automatic scaling from zero to thousands of concurrent users with no infrastructure management. The system's total operational cost is bounded by per-inference API costs, making it highly economical for both prototyping and production deployment.

### 8.2 Failure Modes and Known Limitations

Despite strong user study outcomes, Dayam exhibits several identifiable failure modes:

- **Occluded or ambiguous plating.** When a food item is partially obscured (e.g., a dish under heavy sauce or garnish), Stage 1 may undercount ingredients. In the field study, this manifested primarily in street food (contributing to the 78% accuracy rating in that category) and heavily sauced restaurant dishes.
- **Hyperlocal or regional ingredients.** The LLM's parametric knowledge is biased toward West-ern culinary traditions. Obscure regional ingredients — such as *ghongura* (a sour leafy vegetable) or certain fermented fish pastes used in Southeast Asian cuisines — were occasionally misidentified or omitted.
- **Processing-derived non-vegan contamination.** The system does not detect manufacturing cross-contamination (e.g., "may contain milk") or non-vegan processing aids not listed on labels, as these are not visually inferable. This is an inherent limitation of vision-only analysis rather than a system-specific failure.
- **Hash collision robustness.** The MD5 caching scheme is not collision-resistant for security

purposes, though the probability of accidental collision in a food scanning context is negligible (< 1 in $2^{64}$ for random images). A production system should use SHA-256 for cache key generation.

### 8.3 Ethical Considerations

The system's prompting strategy embeds a conservative classification bias — favoring false negatives (marking vegan food as potentially non-vegan) over false positives (marking non-vegan food as vegan). For individuals with strong ethical convictions about animal products, a missed non-vegan ingredient represents a more consequential error than an unnecessary caution alert.

User images are not persistently stored; only the MD5 hash and the structured analysis result are retained in the database. This architectural choice protects user privacy by preventing reconstruction of dietary histories from stored image data.

### 8.4 Broader Impact

Dayam contributes to the broader sustainability agenda by lowering the friction associated with plant-based dietary transitions. Research has established that a global shift toward plant-based diets could reduce food-related greenhouse gas emissions by up to 70% [21]. By providing instant, actionable dietary guidance, systems like Dayam can accelerate this transition. Beyond individual users, the same pipeline architecture could be adapted for other dietary compliance use cases, including halal/kosher verification, allergen detection for food-allergic consumers, and diabetic glycemic index estimation — all from a single food photograph.

## 9. Conclusion and Future Work

This paper presented Dayam, a novel vision-based artificial intelligence framework for real-time vegan diet compliance verification and sustainable alternative recommendation. The proposed system distinguishes itself through the integration of multimodal vision-language models with ontology-driven nutritional reasoning within a unified three-stage pipeline. Unlike existing approaches that rely on isolated food recognition or manual input, Dayam enables ingredient-level analysis, strict compliance verification, and context-aware substitution directly from food images.

A key contribution of this work lies in the incorporation of a curated ontology of 213 animal-derived substances, combined with a carefully designed prompt engineering strategy to enhance hidden ingredient detection and improve classification reliability. Furthermore, the system extends beyond detection by offering nutritionally informed, practical, and actionable plant-based alternatives, thereby supporting real-world dietary decision-making.

The framework is implemented as a serverless edge-based application, achieving a mean end-to-end latency of 3.2 seconds for uncached queries and under 200 milliseconds for cached responses. Empirical validation through a 14-day in-the-wild user study involving 50 participants and 642 food scans demonstrates strong usability and effectiveness, with a System Usability Scale (SUS) score of 86.4, an alternative acceptance rate of 82%, and a significant 14× reduction in dietary verification time compared to manual approaches.

Despite these promising results, several avenues exist for further enhancement. First, large-scale precision and recall benchmarking using a curated, human-annotated dataset of diverse global cuisines is essential to establish objective performance metrics. Second, the integration of personalized dietary profiles, including allergies, intolerances, and regional preferences, can improve the relevance and adaptability of recommendations. Third, extending the framework to support multi-dish and menu-level analysis through advanced segmentation techniques will broaden its applicability in real-world dining scenarios. Finally, on-device model distillation represents a critical step toward enabling offline functionality, reducing latency, and improving accessibility in low-connectivity environments.

In summary, Dayam advances the state of the art by bridging the gap between visual food understanding and intelligent dietary decision support, offering a scalable and practical solution for promoting sustainable and informed food choices.

**References**

[1] J. Poore and T. Nemecek, "Reducing food's environmental impacts through producers and consumers," *Science*, vol. 360, no. 6392, pp. 987–992, Jun. 2018.

[2] Grand View Research, "Vegan Food Market Size, Share & Trends Analysis Report, 2023–2030," Grand View Research, San Francisco, CA, Tech. Rep. GVR-4-68038-955-2, 2023.

[3] F. Biermann and M. Kahlmeier, "Why going vegan is hard: Understanding barriers to plant-based diets," *Appetite*, vol. 164, p. 105260, Sep. 2021.

[4] T. Key, P. Appleby, and M. Rosell, "Health

effects of vegetarian and vegan diets," *Proceedings of the Nutrition Society*, vol. 65, no. 1, pp. 35–41, Feb. 2006.

[5] M. G. Onorati and G. G. Bonetti, "Digital food rating, caring dietary styles, and identity: a study of plant-based restaurant reviews," *Journal of Cultural Economy*, vol. 18, no. 6, pp. 846–867, 2025.

[6] B. Y. Laing *et al.*, "Accuracy of a smartphone application for estimating energy and macronutrient intake," *JMIR mHealth and uHealth*, vol. 2, no. 2, p. e27, 2014.

[7] M. Bolaños, A. Ferrà, and P. Radeva, "Food ingredients recognition through multi-label learn-ing," in *Proc. Int. Conf. Image Analysis and Processing (ICIAP)*, Springer, 2017, pp. 394–404.

[8] L. Bossard, M. Guillaumin, and L. Van Gool, "Food-101 – Mining discriminative components with random forests," in *Proc. European Conf. Computer Vision (ECCV)*, Springer, 2014, pp. 446–461.

[9] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 2818–2826.

[10] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 770–778.

[11] M. Tan and Q. Le, "EfficientNet: Rethinking model scaling for convolutional neural net-works," in *Proc. Int. Conf. Machine Learning (ICML)*, PMLR, 2019, pp. 6105–6114.

[12] OpenAI, "GPT-4 Technical Report," *arXiv preprint arXiv:2303.08774*, Mar. 2023.

[13] H. Liu, C. Li, Q. Wu, and Y. J. Lee, "Visual instruction tuning," in *Proc. Advances in Neural Information Processing Systems (NeurIPS)*, vol. 36, 2023.

[14] Meta AI, "The Llama 4 herd of models," *Meta AI Research Blog*, Apr. 2025. [Online]. Avail-able: https://ai.meta.com/blog/llama-4/

[15] S. M. A. Mahmud *et al.*, "FoodVisor: A Food Calorie Estimation System," in *Proc. IEEE Int. Conf. on Advances in Information Technology (ICAIT)*, 2021, pp. 1–6.

[16] J. Freyne and S. Berkovsky, "Intelligent food planning: Personalized recipe recommendation," in *Proc. Int. Conf. Intelligent User Interfaces (IUI)*, ACM, 2010, pp. 321–324.

[17] C.-Y. Teng, Y.-R. Lin, and L. A. Adamic, "Recipe recommendation using ingredient net-works," in *Proc. ACM Web Science Conf.*, 2012, pp. 298–302.

[18] Z. Yang, E. Khatibi *et al.*, "ChatDiet: Empowering Personalized Nutrition-Oriented Food Recommender Chatbots through an LLM-Augmented Framework," *arXiv preprint arXiv:2403.00781*, Mar. 2024.

[19] S. Khamesian, A. Arefeen, S. M. Carpenter, and H. Ghasemzadeh, "NutriGen: Personalized Meal Plan Generator Leveraging Large Language Models," *arXiv preprint arXiv:2502.20601*, Feb. 2025.

[20] E. Jonas, J. Schleier-Smith, V. Sreekanti *et al.*, "Cloud programming simplified: A Berkeley view on serverless computing," *arXiv preprint arXiv:1902.03383*, Feb. 2019.

[21] M. Springmann, H. C. J. Godfray, M. Rayner, and P. Scarborough, "Analysis and valuation of the health and climate change cobenefits of dietary change," *Proc. National Academy of Sciences*, vol. 113, no. 15, pp. 4146–4151, Apr. 2016.

[22] J. N. Acosta *et al.*, "Multimodal biomedical AI," *Nature Medicine*, vol. 28, pp. 1773–1784, Sep. 2022.

[23] Y. Wu *et al.*, "Can large language models be used to provide dietary advice? A pilot study using GPT-4," *Nutrients*, vol. 15, no. 18, p. 4081, Sep. 2023.

[24] Groq Inc., "Groq API Pricing and Model Documentation," Groq Developer Portal, 2024. [Online]. Available: https://console.groq.com/docs/models

[25] A. Bangor, P. Kortum, and J. Miller, "Determining what individual SUS scores mean: Adding an adjective rating scale," *Journal of Usability Studies*, vol. 4, no. 3, pp. 114–123, 2009.

[26] H.-Y. Yen *et al.*, "A systematic review of the usability of mobile applications for dietary tracking," *Nutrients*, vol. 12, no. 11, p. 3327, Nov. 2020.