WhatTheFridge: Project Report

Project Overview

WhatTheFridge is a web application that identifies ingredients from fridge images and recommends recipes based on available ingredients. The project progressed through multiple technical phases to optimize accuracy, user experience, and performance.

Links:
Deployment: https://whatthefridge.onrender.com/
GitHub repository: https://github.com/sindhu-satish/WhatTheFridge

Problem Statement

Users often have ingredients at home but lack ideas for what to cook. Identifying ingredients from a fridge photo and suggesting recipes could bridge this gap efficiently.

Objectives

- Automatically detect and list ingredients from uploaded fridge images.
- Recommend creative, structured recipes based on detected ingredients.
- Provide a smooth and responsive user experience.

Methodology

Phase 1: Object Detection Approaches

DETR-ResNet Attempt

- Implemented Facebook's DETR model with a ResNet backbone for object detection.
- Used pre-trained weights on the COCO dataset.

Challenges:

- COCO dataset provided limited coverage for food categories.

- The model failed to detect specific ingredients reliably.
- Significant post-processing would have been required to adapt general object detection outputs to ingredient identification.
- Fine-tuning DETR on a specialized food dataset was identified as a potential, but resource-intensive, improvement.

Depth Segmentation Attempt

- Applied depth segmentation to separate overlapping items.
- Combined depth information with object detection to improve identification accuracy.

Challenges:

- Difficulty handling transparent containers and reflective surfaces typical of fridges.
- Increased computational overhead without significant accuracy improvement.
- Complex fridge layouts reduced segmentation effectiveness.

Phase 2: Vision-Language Model Integration

OpenAI GPT-4 Vision API

- Transitioned to using OpenAI's GPT-4 Vision API for direct image analysis.
- Built structured JSON outputs listing detected ingredients, quantities, and confidence scores.

Results:

- Higher precision and reliability in ingredient detection.
- Natural language capabilities allowed better handling of real-world, unstructured fridge photos.
- Significant reduction in required post-processing.

Phase 3: Recipe Recommendation System

GPT-4 Text Model

- Integrated GPT-4 to generate recipes based on selected ingredients.
- Structured recipe outputs including preparation time, cooking time, and step-by-step instructions.

Results:

- Creative and relevant recipes based on available ingredients.
- Consistent formatting allowed easy frontend parsing and display.

Phase 4: Frontend Development

React + TypeScript Application

- Built responsive UI using React, TypeScript, and TailwindCSS.
- Implemented drag-and-drop image upload, ingredient selection, and real-time feedback mechanisms.

Features:

- Clean, responsive design for mobile and desktop users.
- Loading indicators, error handling, and empty state management for better user experience.

Technical Challenges and Solutions

API Integration

- Solved CORS issues and HTTPS mixed content errors.
- Implemented robust timeout and error handling for API interactions.
- Structured backend responses for strict type safety on the frontend.

Backend Processing

- Improved reliability and consistency of image analysis outputs.
- Implemented structured JSON validation and fallback error handling.

User Experience Enhancements

- Introduced loading states for all asynchronous operations.
- Enhanced error messaging for better clarity.
- Built an intuitive ingredient selection and deselection interface.

Current Capabilities

- Upload fridge or food images via drag-and-drop.
- Detect multiple ingredients with estimated quantities and confidence scores.
- Allow users to select detected ingredients.
- Generate creative and structured recipes based on user selections.
- Provide clear, real-time feedback at each stage.

Technologies Used

- **Frontend:** React, TypeScript, TailwindCSS
- **Backend:** FastAPI, Python
- **APIs:** OpenAI GPT-4 Vision API, GPT-4 Text API
- **Deployment:** Render

Future Work

Here are some ideas that could be implemented to improve user experience:

- Record a detection date for each identified food item and alert the user when items are nearing spoilage. This feature aims to help users minimize food waste.
- Implement ingredient categorization (e.g., dairy, vegetables, condiments).
- Enable user accounts for saving preferences and favorite recipes.
- Add support for dietary restrictions and preference-based filtering.
- Improve recipe filtering options (e.g., preparation time, meal type).
- Enhance social sharing capabilities.