TEAM E2

NutrientMatch

Surya Chandramouleeswaran, Grace Liu, Steven Zeng

Use Case

PROBLEM

- An increase in physical wellness trends leads to more people tracking their daily macronutrient intake
- Food tracking becomes a tedious task that often results in lack of consistency

SOLUTION

NutrientMatch is a product that helps users simplify their food tracking process

DESIGN CHANGES

- Calorie tracking only instead of all macronutrient count
- Separating label reading and image classification components
- Arduino driver/receiver instead of camera working with scale

Quantitative Design Requirements

Overview

Structural Parameters

Description of Physical Features of Product

- Wooden cubic structure of edge length **~2-3 feet**
- Ambient **55-80 watt** LED for diffuse illumination
- Scale, Microcontroller

Performance

Object recognition and performance

Captured image of product should be reasonably clear and well-illuminated

- **2 MP, 15 FPS** Camera
- Food held **12-18 inches** from camera lens
- **~750 pixel images** to feed into CNN

Accessibility

User Accessibility and Experience

Not a quickly-patched DIY; product should reflect elegance and functionality

- Overall Latency minimization (**~25 seconds**)
- Systematic Bias Mitigation (**>= 90%** recognition accuracy)

Evolution of Solution Approach

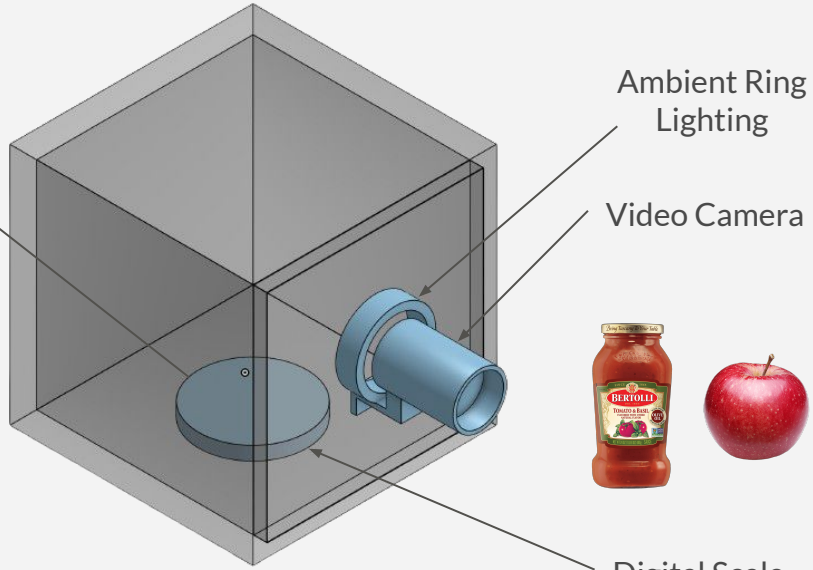
Themes	Object Recognition(CNN)	Back-end Management	Hardware Considerations
STEP 1	Image Classification: -> Packaged Items -> Standalone Foods	mySQL manages users within family, corresponding data	Diffuse Lighting Structure
STEP 2	Label Reading: -> Directed towards packaged foods	Low-latency (25-seconds) updating process	Scale Communication: Arduino RS232/RPI/ESP8226
STEP 3	Robust Error Handling	“Polling strategy” on weight measurement	Scripts for microchip control

Societal/Ethical Considerations

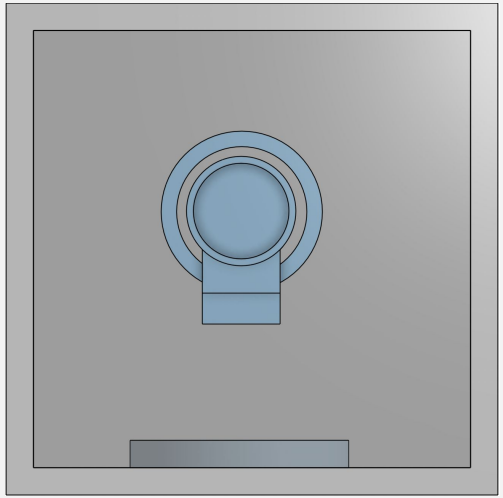
- Mitigating bias in the classification algorithm
- Maintaining health nutritional intake can be challenging for those with busy lifestyles (has to be reasonably efficient)
- Food wastage concerns

Solution Approach: Overview

ESP8266



Isometric View

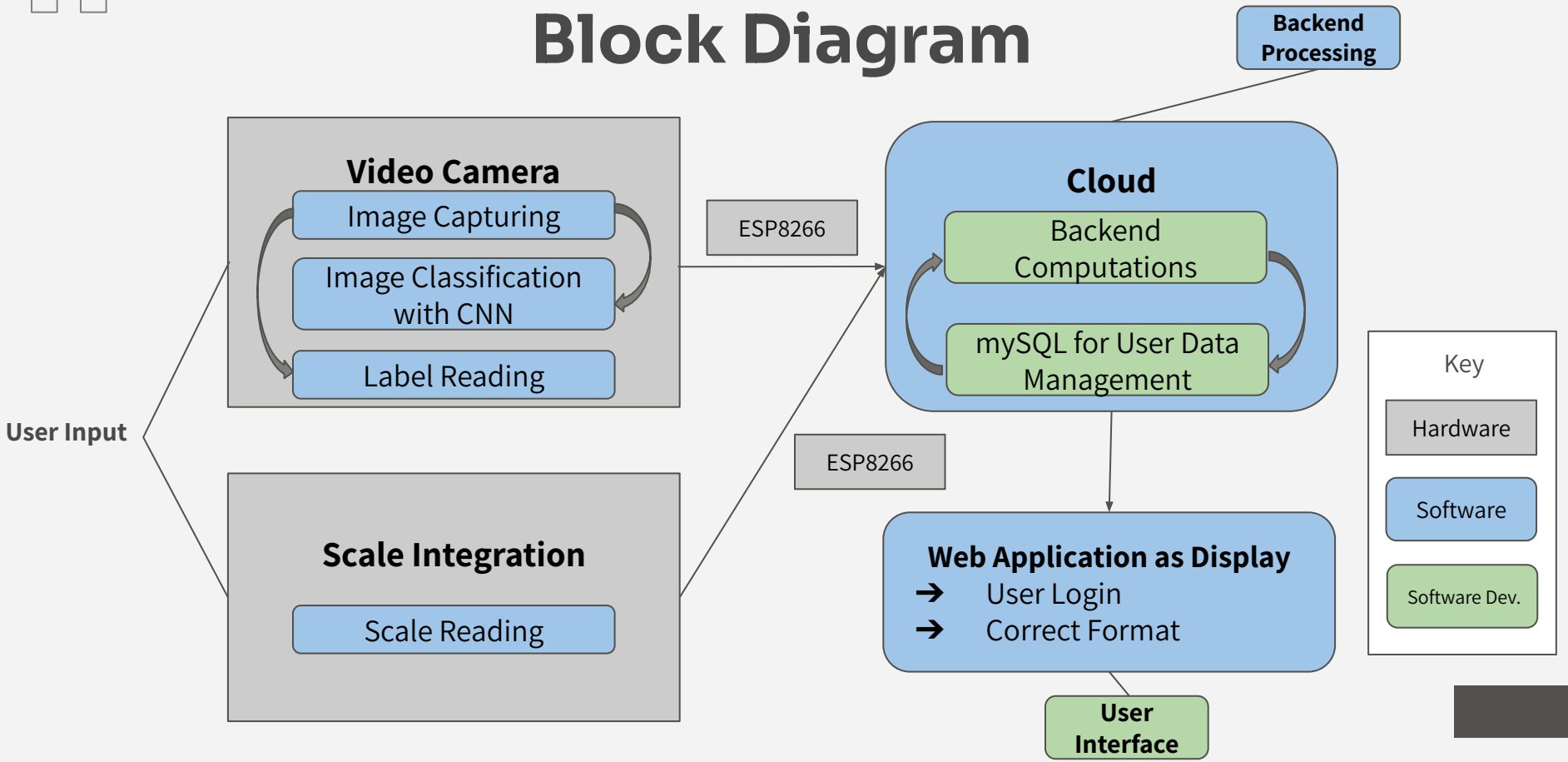


Front View

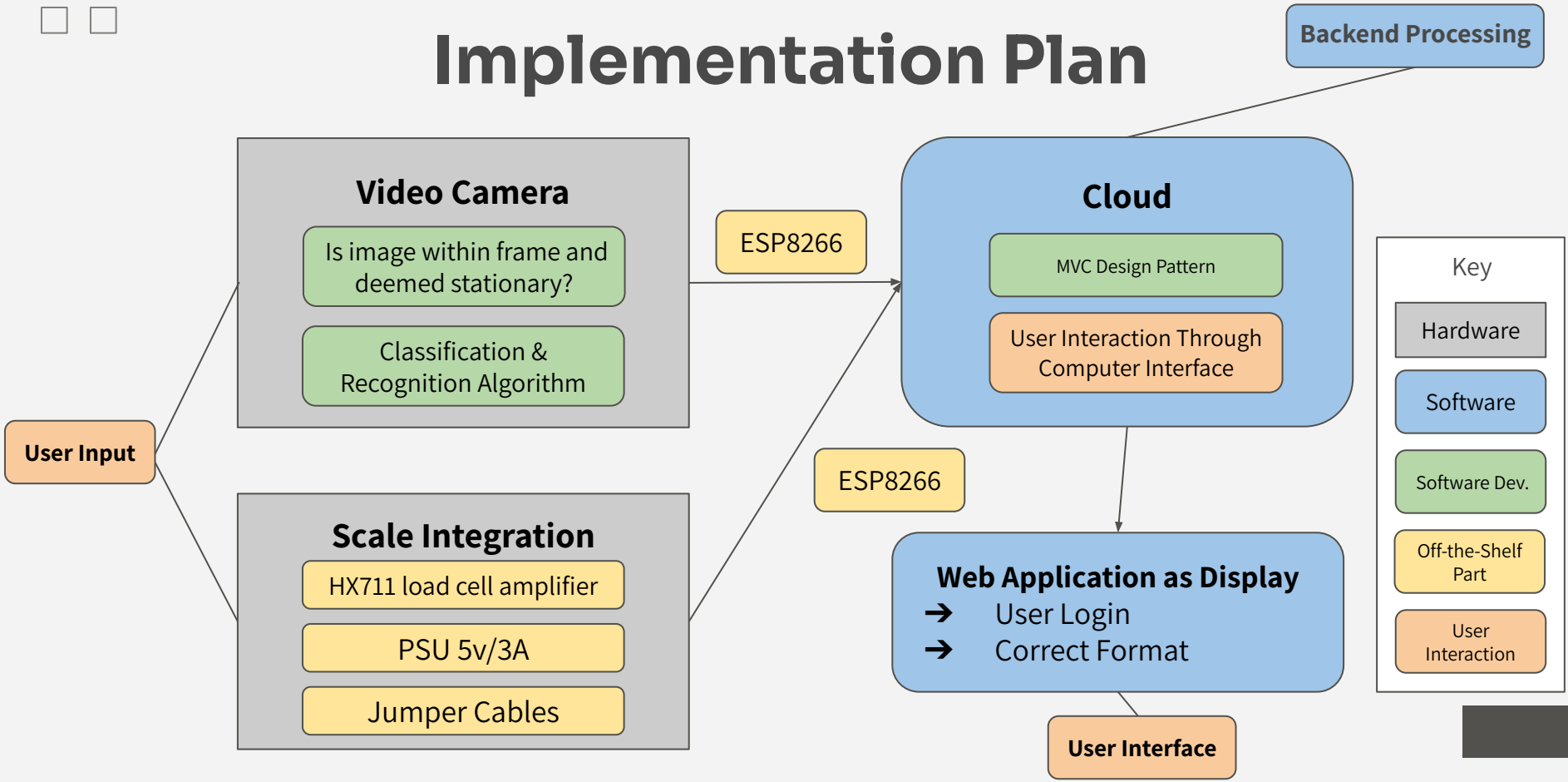
Users can either choose to

- Scan nutritional label on packaged foods
- Image classify & weigh fresh produce on the scale

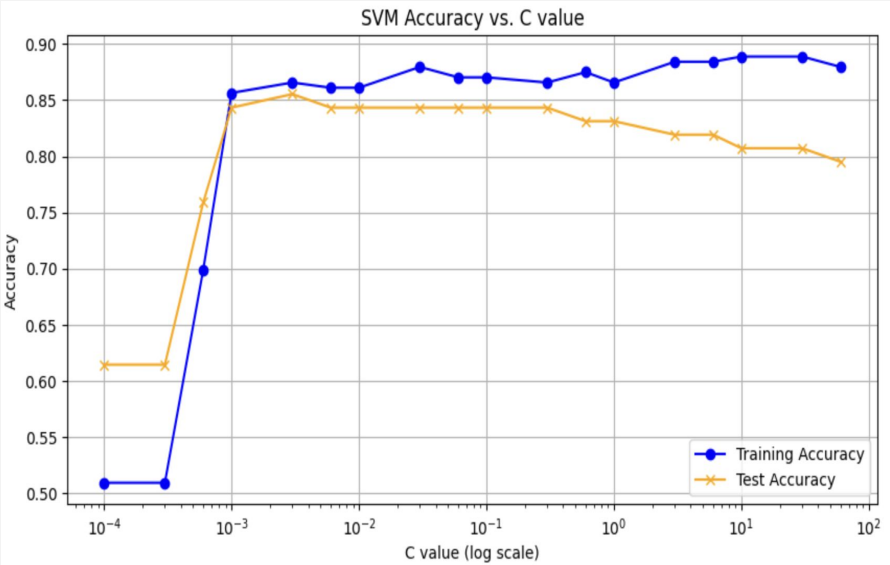
Block Diagram



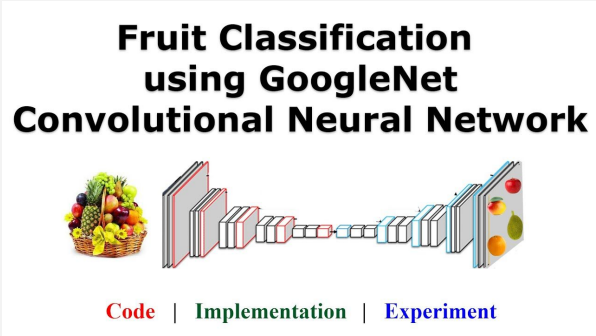
Implementation Plan



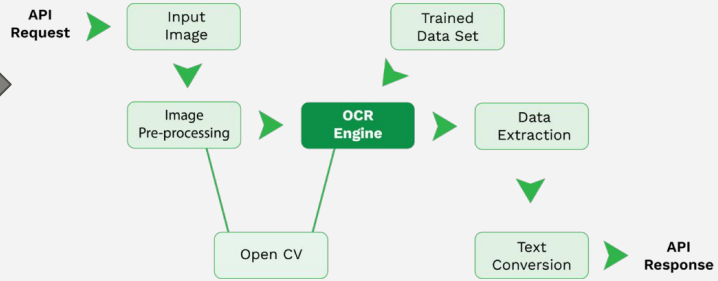
Design Progress



SVM training set with images to classify between raw and packaged items



22-Layer GoogleNet Model to Classify Raw Foods

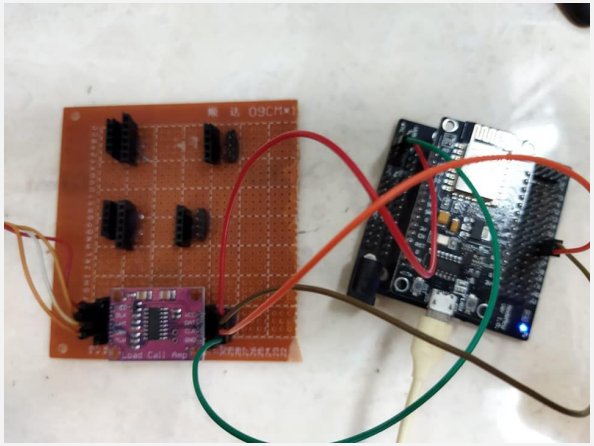


Tesseract API Call (label reading)

From: https://www.youtube.com/watch?app=desktop&v=58-1KmsIEcQ

Scale Design

- Requires accurate measurement and timely communication
  - Important to start thinking about pertinent issues now
- “Polling Feature” requires control hardware to manage communication between scale and database
- Most scales have a USB/COM port
  - Plan is to hook this to an ESP8266 WiFi Microchip (proof of concept)
  - Anticipate Load-Cell Amp will be needed to enhance signal strength (pictured right)
- Xiaomi Smart Scale with RPI
  - Seamless Bluetooth integration with mySQL (stretch goal)



Change in voltage in load cell too small to be captured by ADC: hence need for amplifier

Testing, Verification, Validation

01 **Image Recognition System**

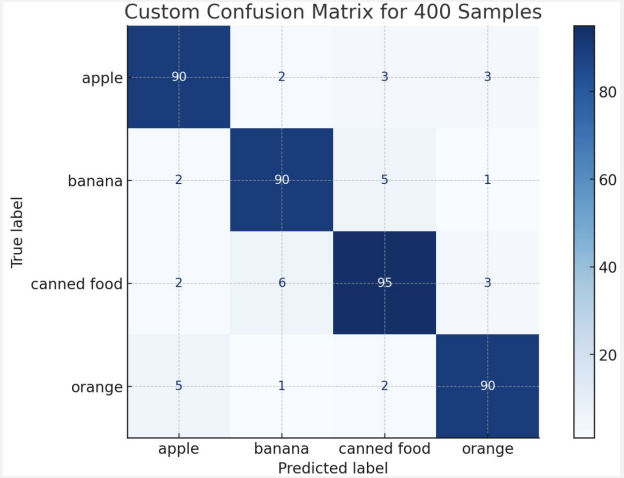
- Canned foods/snacks
- Whole Foods
- Invalid Groups

02 **Overall Performance Considerations**

- Weight detection and update
- Overall Process Latency

03 **Risk Mitigation: Invalid Measurement**

- Ability to interrupt/restart measuring process in flexible manner



- Benchmark: 95% category classification
- After classification, caloric values should be within 10% range from actual amount

Goal: Modular structure around imaging, scale integration, and backend management

Schedule

Steven Surya Grace **All Three of Us**