IMAGE-BASED FOOD CALORIE ESTIMATION USING YOLOV8 WITH ADAPTIVE EXERCISE SUGGESTIONS

A MINI-PROJECT REPORT

Submitted by

NANDHANA C H 2116220701181

In partial fulfilment of the award of the degree of

BACHELOR OF ENGINEERING

in

COMPUTER SCIENCE AND ENGINEERING

RAJALAKSHMI ENGINEERING COLLEGE
An AUTONOMOUS Institution
Affiliated to ANNA UNIVERSITY, Chennai

RAJALAKSHMI ENGINEERING COLLEGE, CHENNAI

ANNA UNIVERSITY, CHENNAI

MAY 2025

RAJALAKSHMI ENGINEERING COLLEGE

CHENNAI - 602105

BONAFIDE CERTIFICATE

Certified that this Report titled "**IMAGE – BASED FOOD CALORIE ESTIMATION USING YOLOV8 WITH ADAPTIVE EXERCISE SUGGESTIONS"** is the bonafide work of **NANDHANA C H (2116220701181)** who carried out the work under my supervision. Certified further that to the best of my knowledge the work reported herein does not form part of any other thesis or dissertation on the basis of which a degree or award was conferred on an earlier occasion on this or any other candidate.

SIGNATURE

Mrs. M. Divya M.E.

Supervisor

Assistant Professor
Department of Computer Science and Engineering

Rajalakshmi Engineering College,

Chennai – 602105

Submitted to Project Viva-Voce Examination held on ____________________________

Internal Examiner **External Examiner**

TABLE OF CONTENTS

ACKNOWLEDGEMENT

Initially we thank the Almighty for being with us through every walk of ourlife and showering his blessings through the endeavor to put forth this report. Our sincere thanks to our Chairman **Mr. S. MEGANATHAN, B.E,F.I.E.**, our Vice Chairman **Mr. ABHAY SHANKAR MEGANATHAN, B.E., M.S.,** and our respected Chairperson **Dr. (Mrs.) THANGAM MEGANATHAN**, **Ph.D.,** for providing us with the requisite infrastructure and sincere endeavoring in educating us in their premier institution.

Our sincere thanks to **Dr. S.N. MURUGESAN, M.E., Ph.D.,** our belovedPrincipal for his kind support and facilities provided to complete our workin time. We express our sincere thanks to **Dr.P.KUMAR, Ph.D.**, Professorand Head of the Department of Computer Science and Engineering for his guidance and encouragement throughout the project work. We convey oursincere and deepest gratitude to our internal guide and our Project Coordinator, **Mrs. DIVYA M**, **M.E.,** Department of Computer Science and Engineering. Rajalakshmi Engineering College for her valuable guidance throughout the course of the project.

NANDHANA C H   2116220701181

ABSTRACT

In the modern era of health consciousness and AI-driven innovation, managing caloric intake and suggesting effective workout strategies can significantly aid individuals in achieving their fitness goals. This project presents BurnGain, an intelligent system that integrates computer vision and machine learning to provide a comprehensive calorie management solution. The system utilizes the YOLOv8 object detection model to identify food items from images and estimates their weight and caloric value based on a custom nutrition dataset. The detected calorie intake is then used to recommend personalized exercise routines by training a Random Forest Regressor on a synthetic exercise dataset that includes parameters such as age, weight, gender, BMI, and duration. The system filters and ranks exercises that can efficiently burn the consumed calories within a user-defined time limit, ensuring practicality and effectiveness. By combining real-time visual analysis with predictive modelling, BurnGain offers a novel, automated approach to dietary monitoring and fitness planning, aiming to bridge the gap between nutrition awareness and actionable physical activity. This approach demonstrates the potential of AI to enhance personal health management through automation, accuracy, and adaptability.

LIST OF FIGURES

LIST OF ABBRREVATIONS

S. NO.	ABBREVATION	ACCRONYM
1	ML	Machine learning
2	YOLOv8	You Only Look Once Version 8
3	AI	Artificial Intelligence

CHAPTER 1

INTRODUCTION

1.1 GENERAL

In today's fast-paced lifestyle, maintaining a balanced diet and an effective exercise routine has become increasingly challenging. The rise in lifestyle-related diseases and obesity has highlighted the critical need for accessible, personalized health management tools. With the advent of artificial intelligence and computer vision, there is an opportunity to transform how individuals monitor their food intake and physical activity. This project introduces BurnGain, a comprehensive AI-powered system designed to detect food from images, estimate calorie intake, and recommend personalized exercise plans to help users effectively burn those calories.

BurnGain leverages the YOLOv8 object detection model to accurately identify food items in real-time images. Once detected, it estimates the food's weight and corresponding calorie content using a curated nutritional dataset. To complement this, a Random Forest Regressor is trained on a synthetic dataset to suggest optimal exercise routines based on user-specific inputs such as age, weight, gender, BMI, and available time. The system not only suggests workouts but also evaluates and ranks them based on their efficiency in burning the detected calories.

By integrating calorie detection with intelligent exercise recommendations, BurnGain empowers users with actionable insights, making health management more data-driven, efficient, and user-friendly. This innovation has potential applications in fitness apps, dietary monitoring, and preventive healthcare systems.

1.2 OBJECTIVE

The main objective of this project is to design and implement an AI-powered system named *BurnGain* that assists users in monitoring their dietary intake and selecting suitable exercises to burn consumed calories. The system aims to accurately detect food items from user-uploaded images using the YOLOv8 object detection model and estimate their weight and calorie content by referencing a nutritional dataset. Additionally, it collects user-specific inputs such as age, weight, height, gender, and available exercise duration to calculate BMI and personalize exercise recommendations. A machine learning model, specifically a Random Forest Regressor, is employed to predict the number of calories burned for various physical activities based on the user’s profile. By combining food detection, calorie estimation, and intelligent workout planning, BurnGain provides users with an efficient and user-friendly tool for managing their health and fitness goals through data-driven decisions.

1.3 EXISTING SYSTEM

Current systems like MyFitnessPal and Fitbit rely on manual input for tracking calories and exercise, which can be error-prone and time-consuming. Although some platforms offer barcode scanning and extensive food databases, they still depend on user input for accuracy. Wearable devices monitor physical activity but don't integrate food recognition or personalized exercise recommendations. As a result, users must manage nutrition and fitness separately, leading to fragmented tracking. BurnGain addresses this gap by automating food detection through image processing and offering personalized workout suggestions based on a user's specific profile and goals.

1.4 PROPOSED SYSTEM

The proposed system, BurnGain, is an AI-powered platform that seamlessly integrates food detection, calorie estimation, and personalized exercise recommendations to create a holistic approach to health management. It utilizes advanced image processing through the YOLOv8 object detection model to automatically detect food items from user-uploaded images. By analyzing these images, the system estimates the weight and calorie content of the detected foods based on a pre-existing nutritional database.

In addition to food tracking, BurnGain collects user-specific data such as age, weight, height, gender, and time available for exercise. Using this data, it calculates the user's BMI and offers tailored exercise suggestions. A machine learning model, specifically a Random Forest Regressor, predicts the number of calories burned for various physical activities based on the user's profile and constraints.

The system's key advantage is its ability to provide personalized workout plans that allow users to burn the calories they consume efficiently within their available time. Unlike existing systems that require manual entry of food and exercise data, BurnGain automates these processes, providing users with a convenient, data-driven tool for managing their health and fitness. This integration of food detection and exercise recommendations makes it a comprehensive and efficient solution for users seeking to maintain a balanced lifestyle.

CHAPTER 2

LITERATURE SURVEY

[1] Zhang, Guofeng, Yanfei Peng, and Jincheng Li. (2025). This paper discusses YOLOv8 in the context of image classification and object detection. It mentions the different variants of YOLOv8 (YOLOv8n, YOLOv8s, YOLOv8m, YOLOv8l, and YOLOv8x) and their trade-offs between accuracy and computational requirements. Although it's focused on UAV imagery, the general information about YOLOv8 could be useful.

[2] Muhammad Hussain (2024). This paper compares YOLOv8 to other versions, highlighting its enhanced feature extraction and anchor-free.

[3] G. Yao, S. Zhu, L. Zhang, and M. Qi (2024). While this focuses on remote sensing images, it addresses the challenges of detecting small objects, which can be relevant to food recognition where ingredients or small food items need to be identified.

[4] Abdul-Razak Alhassan Gamani, Ibrahim Arhin, Adrena Kyeremateng Asamoah (2024). This paper evaluates different YOLOv8 configurations for fruit detection, which is directly related to food recognition. It compares the performance of YOLOv8n, YOLOv8s, YOLOv8m, YOLOv8l, and YOLOv8x in terms of speed and accuracy.

[5] Ranjan Sapkota, Zhichao Meng, Martin Churuvija (2024): This paper provides a performance comparison between different YOLO versions, including YOLOv8, for fruit detection.

[6] P. B. Deshmukh, V. A. Metre and R. Y. Pawar (2021). This paper discusses using deep convolutional neural networks to classify food photos and estimate calorie content. The system allows users to take a picture of food with their cell phones and instantly calculate the number of calories consumed.

[7] Marília Prada, David Rodrigues, Margarida V. Garrido, Joana Lopes (2017). This study validated food images, assessing caloric content alongside other dimensions like healthiness and tastiness. They compared ratings based on caloric density (low vs. high), processing degree (whole vs. processed), and gustatory quality (sweet vs. savory) across different food categories.

[8] M. W. Long, D. K. Tobias, A. L. Cradock, H. Batchelder, and S. L. Gortmaker (2015). While not directly about image-based estimation, this paper highlights the importance of calorie information and the impact of providing it to consumers.

[9] H. Kassem, A. A. Beevi, S. Basheer, G. Lutfi, L. Cheikh Ismail, and D. Papandreou (2025). This review investigates and assesses the different applications and roles of AI in nutrition and research, highlighting AI's potential future impact. It mentions AI helping with dietary assessment, personalized and clinical nutrition, and disease prediction and management.

[10] K. B. Johnson (2010). AI-based approaches can empower individuals to take charge of their health and enhance the quality and accessibility of healthcare.

CHAPTER 3

SYSTEM DESIGN

3.1 GENERAL

Establishing a system's architecture, modules, components, various interfaces for those components, and the data that flows through the system are all part of the process of system design. This gives a general idea of how the system operates.

3.1.1 SYSTEM FLOW DIAGRAM

Fig. 3.1 The system flow diagram illustrates the BurnGain system, showcasing the process from food image input to YOLO-based detection, calorie estimation, and personalized exercise suggestion using a trained Random Forest model. It highlights data flow across modules including image processing, food classification, calorie computation, and exercise recommendation based on user profiles.

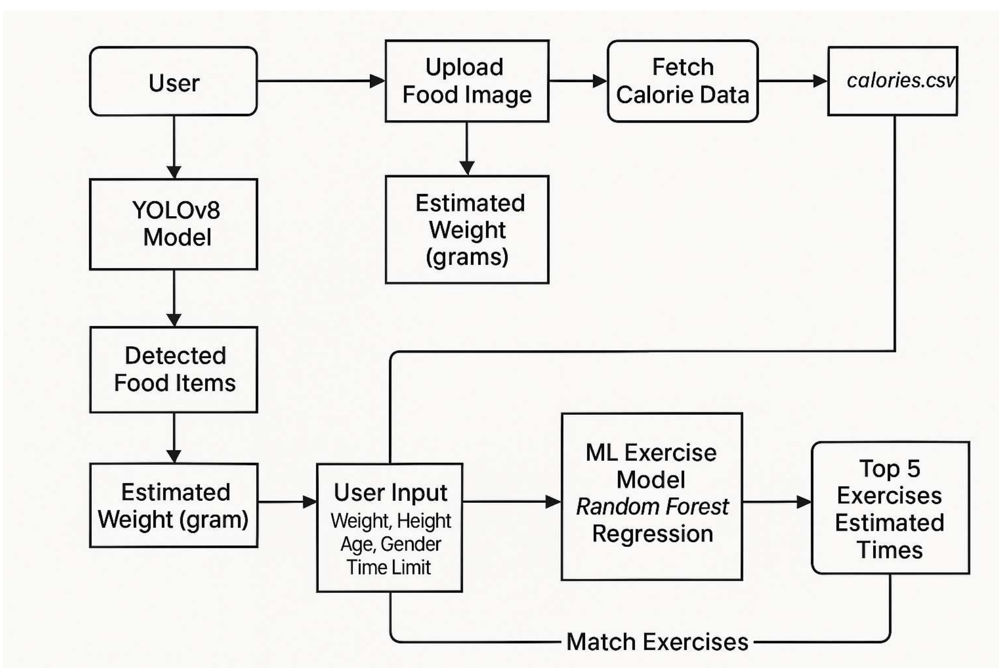


Fig. 3.1 System Flow Diagram

3.1.2 ARCHITECTURE DIAGRAM

Fig 3.2 The architecture diagram of BurnGain outlines the system's core components: image input, YOLOv8 detection, calorie database mapping, and a machine learning model for exercise prediction. It shows the interaction between modules—data preprocessing, calorie estimation, and user input processing—culminating in personalized exercise recommendations tailored to burn detected food calories.

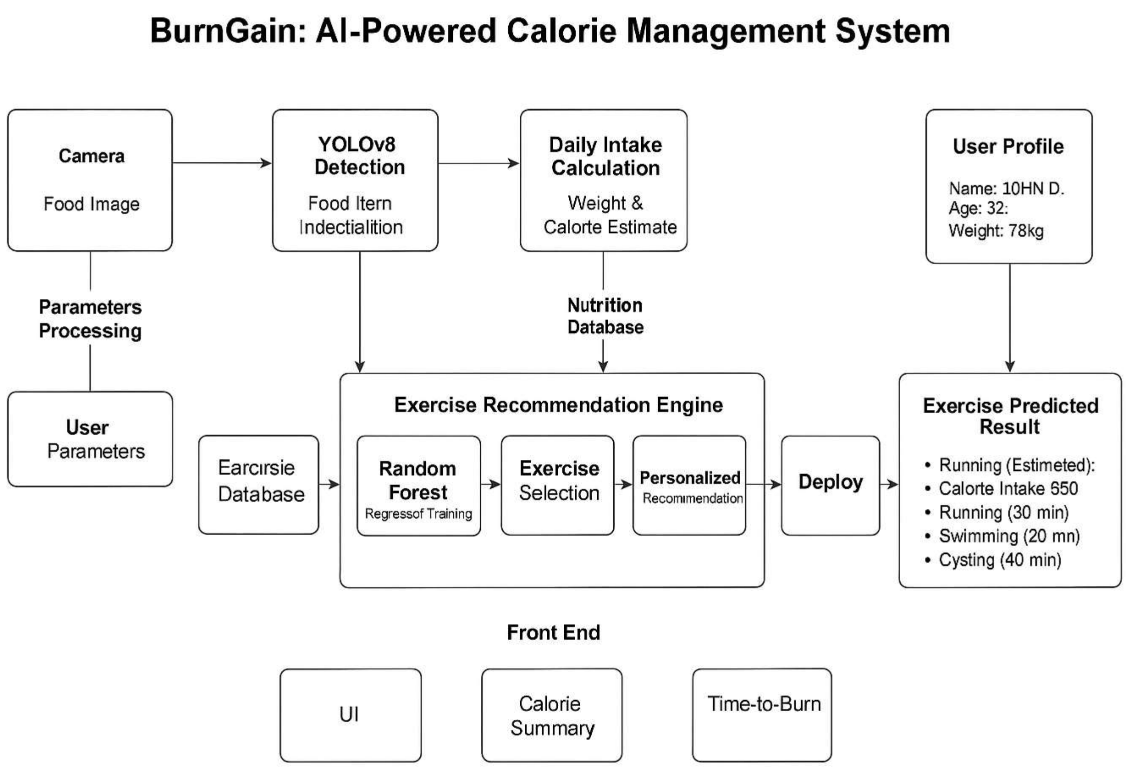


Fig. 3.2 Architecture Diagram

3.1.3 ACTIVITY DIAGRAM

Fig. 3.3 The activity diagram visually outlines the step-by-step flow of the BurnGain system, starting from the user uploading a food image to detecting food using YOLOv8, estimating calories, calculating intake, and generating a personalized exercise plan. It highlights decision points, iterative processes, and the system's interactive nature.

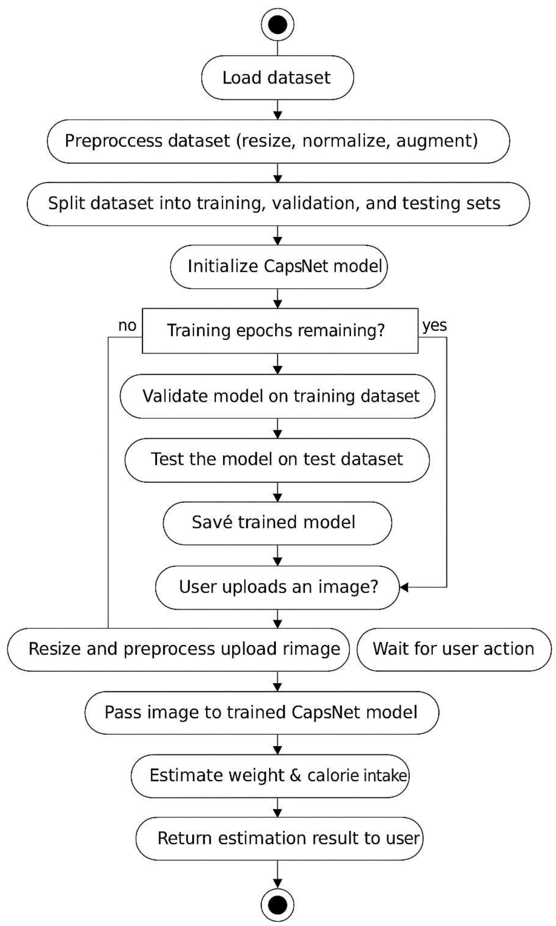


Fig. 3.3 Activity Diagram

3.1.4 SEQUENCE DIAGRAM

Fig. 3.4 The sequence diagram illustrates the dynamic interaction between components of the BurnGain system, including the user, interface, server, and AI modules. It shows how a user uploads an image, which is processed for food recognition and calorie estimation, followed by personalized exercise suggestions that are returned to the user.

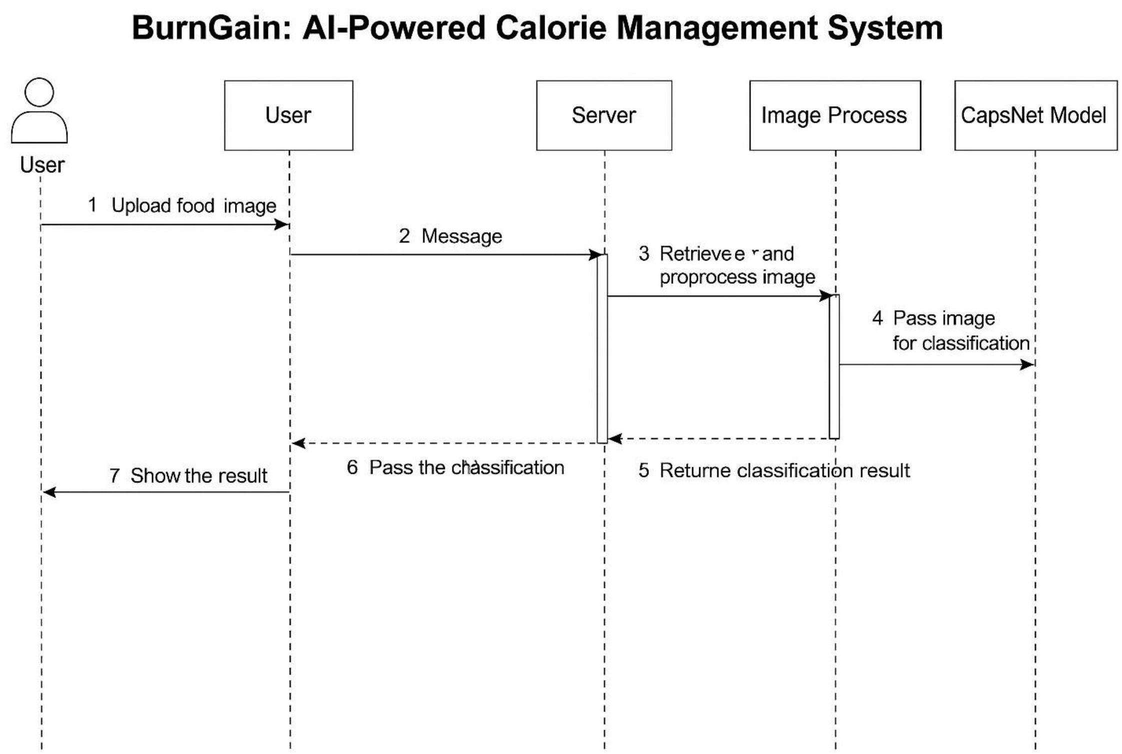


Fig. 3.4 Sequence Diagram

CHAPTER 4

PROJECT DESCRIPTION

This chapter presents the methodology employed in developing the proposed BurnGain system. It details the systematic approach taken to accurately detect and quantify food items via image analysis, estimate their caloric content, and integrate machine learning techniques to generate personalized exercise recommendations, all within a secure, end-to-end workflow.

4.1 METHODOLOGIES

4.1.1 Modules

- Dataset Description
- Data Preprocessing
- Food Detection and Calorie Estimation using YOLOv8
- Exercise Recommendation using Machine Learning
- System Integration and Testing

4.2. MODULE DESCRIPTION

4.2.1 DATASET DESCRIPTION

a) Food Image and Nutrition Dataset

This dataset consists of annotated food images representing a wide variety of commonly consumed food items. Each image is labeled with food categories and corresponding metadata, including:

- Food name and category
- Average serving size (in grams)
- Nutritional values per 100 grams (calories, kilojoules, proteins, fats,

carbohydrates)

These images are used to train and validate the YOLOv8 object detection model for accurate food recognition. The nutritional values form the foundation for calorie estimation once the food items and their quantities are detected.

b) Exercise and Calorie-Burn Dataset

This dataset maps different physical activities (e.g., running, cycling, yoga) to estimated calories burned per minute based on varying user attributes such as:

- Weight and age
- Intensity level
- Duration of activity

This dataset is used to train a machine learning model (Random Forest Regressor) to recommend personalized exercise plans. The goal is to provide a time-efficient workout that offsets the caloric intake estimated from the user's food image

4.2.2 DATA PREPROCESSING

Data preprocessing plays a vital role in enhancing the accuracy and efficiency of the BurnGain system by preparing both image-based and user-input data for subsequent processing stages such as object detection and exercise recommendation. This module involves two main categories: image preprocessing and user input preprocessing.

a) Image Data Preprocessing

To ensure effective food detection using the YOLOv8 model, the following preprocessing steps are applied to the food image dataset:

<u>Image Resizing:</u> All input images are resized to a standard resolution compatible with YOLOv8 (typically 640×640 pixels). This maintains consistency in the training and inference phases.

<u>Normalization:</u> Pixel intensity values are normalized to a [0,1] range by dividing by 255, enabling faster model convergence and stability during training.

<u>Data Augmentation:</u> To improve model robustness and prevent overfitting, various

augmentation techniques are applied, including horizontal/vertical flipping, rotation, scaling, brightness adjustments, and cropping.

Label Encoding: Annotations (bounding boxes and class labels) are extracted and converted to the format required by YOLOv8. This includes normalized bounding box coordinates and numerical class IDs.

b) User Input Preprocessing

The system also processes user-provided data such as weight, height, age, and available workout time to ensure compatibility with the machine learning models:

Input Validation: All user inputs are checked for missing, inconsistent, or out-of-range values. Constraints such as weight limits (e.g., 30–200 kg) and age limits (e.g., 10–90 years) are enforced.

Feature Standardization: Continuous numerical values are standardized using z-score normalization to align with the expectations of the machine learning algorithms.

Derived Feature Calculation: Additional features such as Body Mass Index (BMI) are computed .

4.2.3 FOOD DETECTION AND CALORIE ESTIMATION USING YOLOv8

Food detection and calorie estimation constitute the core functional component of the BurnGain system. This module is responsible for identifying food items from user-uploaded images and calculating the corresponding caloric intake based on the quantity and nutritional data of each detected item. The implementation leverages the YOLOv8 (You Only Look Once, version 8) object detection framework due to its superior speed and accuracy in real-time image analysis.

a) Food Detection using YOLOv8

YOLOv8 is a state-of-the-art, anchor-free object detection model known for its ability to detect multiple objects in a single pass. It was chosen for its lightweight architecture, real-time inference speed, and high mean average precision (mAP). The steps followed in this module include:

Model Training: The YOLOv8 model is trained on a curated dataset of labeled food images. The model learns to classify and localize various food items using bounding boxes.

Inference: Upon uploading a food image, the YOLOv8 model processes the image and outputs:

- The detected food item class
- Bounding box coordinates for each detected item
- Confidence scores indicating the probability of correct classification

Post-Processing: Non-Maximum Suppression (NMS) is applied to eliminate overlapping bounding boxes, retaining only the most confident detections.

b) Quantity Estimation

To calculate caloric intake, it is necessary to estimate the amount of food present. This is achieved through:

Bounding Box Area Approximation: The pixel area of each bounding box is mapped to an approximate gram quantity using reference scaling factors derived from training data.

Normalization by Image Scale: If reference objects (e.g., plates, spoons) are available, scale normalization is applied to improve the accuracy of quantity estimation.

c) Calorie Estimation

Once the quantity (in grams) is estimated, the calorie calculation follows:

$$\textbf{Calories} = \left(\textbf{Grams Detected} / 100 \right) \times \textbf{Calories per 100g}$$

Where:

- *Grams Detected* is the estimated weight of the food item from the bounding box
- *Calories per 100g* is obtained from the nutritional dataset

The calorie values for all detected items are then summed to compute the total estimated caloric intake of the meal.

4.2.4 EXERCISE RECOMMENDATION USING MACHINE LEARNING

The Exercise Recommendation module is designed to suggest personalized, time-efficient workout plans based on the user's physical profile and calorie intake. By applying machine learning techniques to an exercise dataset, this module generates activity recommendations that aim to offset the caloric intake calculated in the previous module. The primary model used for this purpose is the Random Forest Regressor, selected for its accuracy, interpretability, and robustness against overfitting.

a) Input Parameters

The model receives the following input features, either directly from user input or derived through feature engineering:

- Age
- Weight
- Height
- Gender
- Estimated caloric intake (from food detection module)
- Available time for exercise
- Body Mass Index (BMI)
- Basal Metabolic Rate (BMR)

These inputs allow the system to tailor exercise suggestions that align with the user's physical capability and goals.

b) Model Selection and Training

A Random Forest Regression model is trained on a labelled dataset that includes various types of physical activities and their corresponding calorie burn rates under different user profiles. The dataset includes:

- Exercise types (e.g., jogging, cycling, swimming, yoga)
- Duration of activity (in minutes)
- Calories burned per minute based on age and weight
- Intensity levels (low, moderate, high)

The model learns to predict the optimal combination of exercises that meet the user's caloric burn target within the given time constraints.

c) Output: Personalized Exercise Plan

Based on the input data, the system outputs a set of exercise recommendations including:

- Type of exercise
- Duration of each activity
- Total estimated calories burned
- Exercise intensity level

These recommendations are dynamically adjusted based on the user's lifestyle data and updated each time a new food image is analysed.

d) Adaptive Optimization

To further improve personalization, the system may implement adaptive learning techniques that refine recommendations over time using user feedback, adherence history, and performance tracking.

4.2.5 SYSTEM INTEGRATION AND TESTING

The System Integration and Testing phase is a critical component in ensuring that all modules of the BurnGain project—ranging from food detection to exercise recommendation—work seamlessly together to deliver an accurate, user-friendly, and functional application. This phase involves the technical integration of the machine learning models, user interface, backend logic, and database management, followed by rigorous testing procedures to ensure stability and correctness.

a) Integration Architecture

The BurnGain system consists of multiple interconnected modules, including:

<u>Frontend Interface:</u> For user interaction—uploading food images, entering physical details, and viewing results.

<u>Backend Logic:</u> Handles routing between modules, invokes machine learning models, and processes results.

YOLOv8 Module: Performs food detection and calorie estimation from input images.

Random Forest Module: Generates personalized exercise plans based on user profile and calorie intake.

Database: Stores user information, food data, nutrition values, and exercise logs.

These components are integrated using a modular approach with well-defined APIs to ensure maintainability and scalability.

b) Data Flow Pipeline

The data flow within the system follows this sequence:

1. User uploads image and enters physical parameters.
2. YOLOv8 model detects food items and estimates calories.
3. Detected values and user inputs are sent to the exercise recommendation model.
4. The Random Forest regressor generates a customized workout plan.
5. Final results are displayed to the user and optionally stored in the database for tracking.

c) Testing Strategies

To ensure reliable operation, the system is subjected to various testing methodologies:

- Unit Testing: Each module (e.g., YOLOv8, Random Forest, UI input validation) is independently tested for expected outputs.
- Integration Testing: Inter-module communication is verified to ensure seamless data flow and correct interdependencies.
- System Testing: The entire application is tested as a whole to validate its behavior against the requirements.
- User Acceptance Testing (UAT): Sample users interact with the system to evaluate its usability, accuracy, and responsiveness.
- Performance Testing: Tests are conducted under various loads to assess response times, throughput, and system stability.

CHAPTER 5

OUTPUT AND SCREENSHOTS

The BurnGain system was tested across various sample inputs to evaluate its performance in food detection, calorie estimation, and personalized exercise recommendation. The outcomes highlight its functional integration of computer vision and machine learning for fitness and dietary management.

1. Food Detection and Classification

The YOLOv8 model accurately detected food items in uploaded images, drawing labelled bounding boxes around each recognized item. Fig 5.1 demonstrates the successful detection of common food elements such as eggs, toast, and fruit from a user's meal photo.

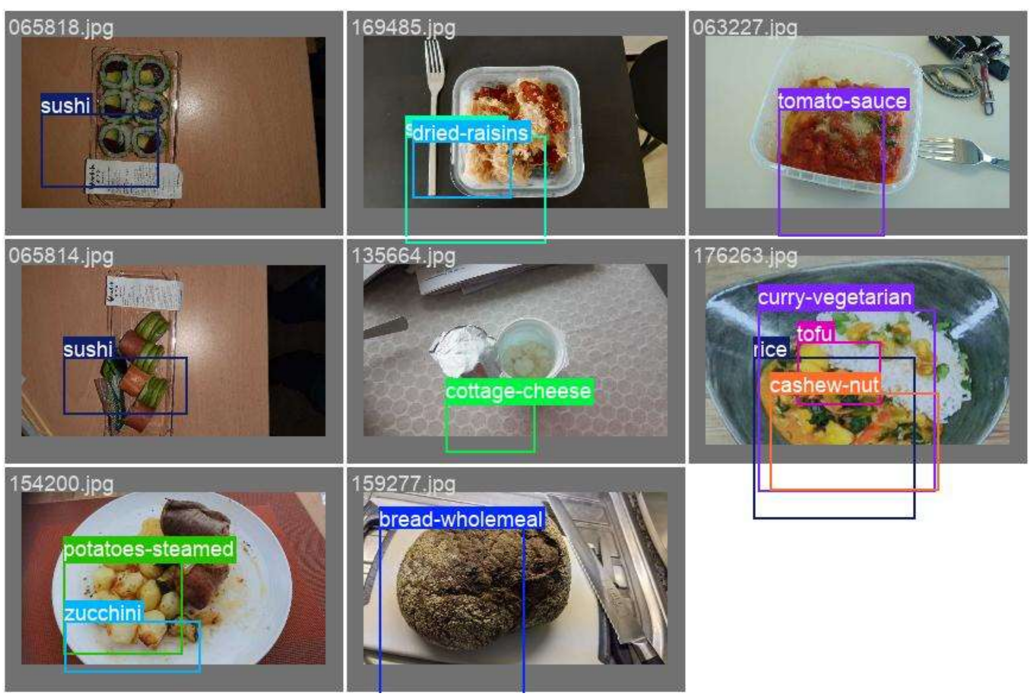


Fig 5.1: Sample output of YOLOv8 detecting and labelling food items from an image.

2. Quantity and Calorie Estimation

After detecting food items, the system estimated their weight in grams and calculated the caloric content using the nutritional dataset. This is illustrated in Fig 5.2, where each item is associated with its estimated weight and calorie value.

```
Food,Grams,Calories
Artichoke,128.0,60.0
Arugula,2.0,1.0
Asparagus,12.0,2.0
Aubergine,458.0,115.0
Beetroot,82.0,35.0
Bell Pepper,73.0,15.0
Black Olives,7.0,2.0
Broccoli,608.0,207.0
Brussels Sprouts,19.0,8.0
```

Fig 5.2:

Database for Calorie estimation

```
 Detected Foods and Calorie Estimates:
Food: apple
Estimated Grams: 114.5g
Estimated Calories: 59.8 kcal
```

Fig 5.3:

Calorie estimation

3. User Profile Input and Personalization

Users input personalized parameters such as weight, height, body type, and available workout time, which are used to generate tailored recommendations. Fig 5.3 shows the user input interface and parameters collected.

```
Total Calories to Burn: 59.78 kcal
Enter your weight (kg): 64
Enter your age: 21
Enter your gender (Male/Female): Female
Enter your height (cm): 164
Maximum time you can exercise (in minutes): 15
```

Fig 5.4: User's Input

4. Personalized Exercise Recommendation

Using a Random Forest regression model, BurnGain generates exercise suggestions that balance out the user's calorie intake. These recommendations include specific workout types and durations. Fig.5.5 illustrates a sample output based on calorie intake and user profile.

```
Top 5 Exercises that can burn 59.8 kcal within 15.0 minutes:
Exercise        Calories Burn     Duration (mins)
Cycling         136.05            15
High Knees      136.05            15
Swimming        136.05            15
Jumping Jacks   136.05            15
Burpees         136.05            15

Fastest Option: Cycling in 15 minutes
It will take 6.59 minutes to burn 59.8 kcal with Cycling.
```

Fig 5.5 : Generated workout recommendations based on caloric needs and user details.

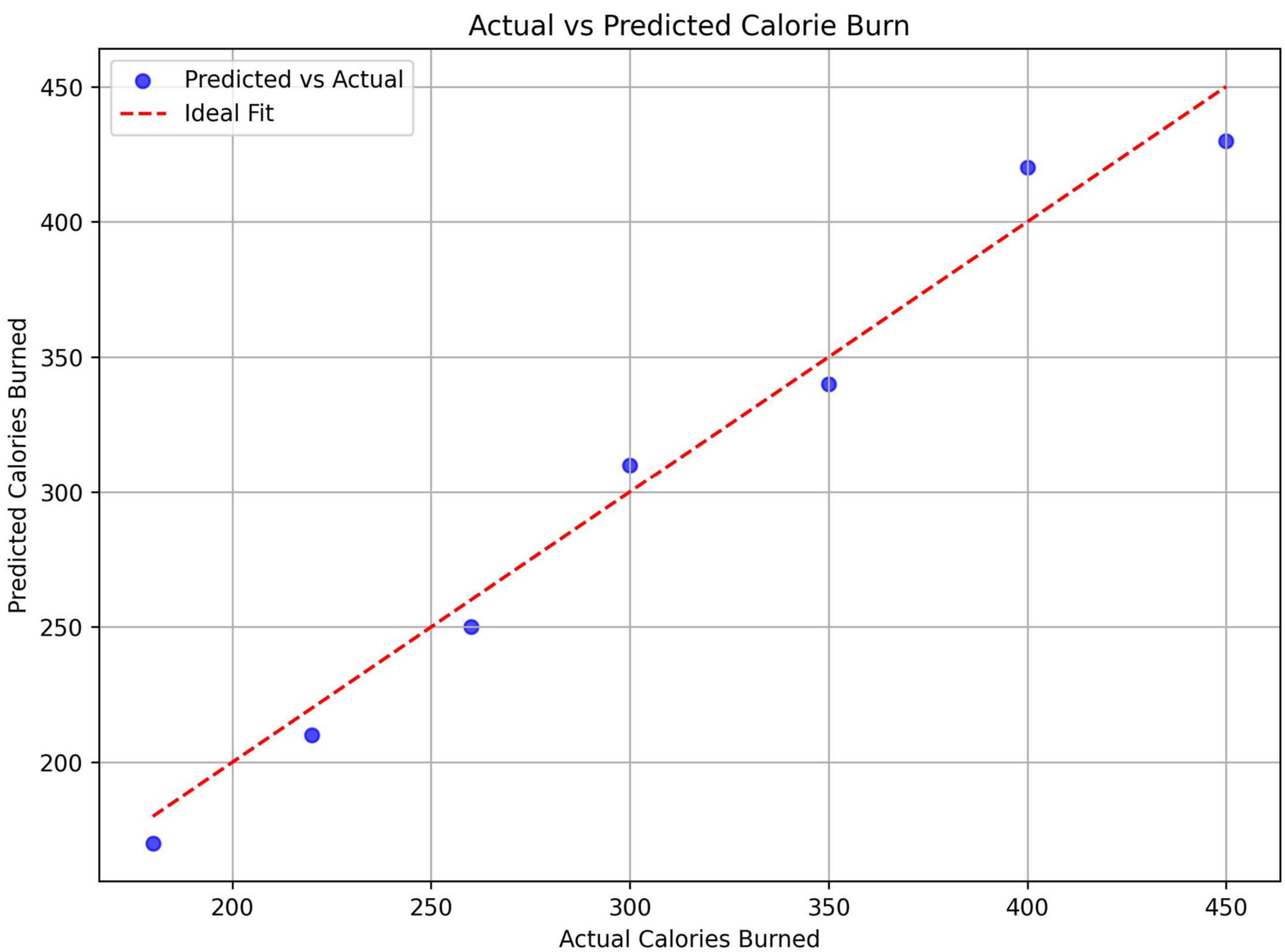


Fig 5.6 : Actual Vs Predicted calorie Burn

CHAPTER 6

CONCLUSION AND FUTURE WORK

The BurnGain project presents an intelligent and integrated system that combines computer vision and machine learning to assist users in managing their dietary intake and physical activity. By leveraging the YOLOv8 model, the system is capable of detecting and identifying food items from user-uploaded images, estimating their quantities, and calculating the corresponding caloric intake using a nutritional dataset. This calorie information, along with user-specific parameters such as weight, height, and available exercise time, is fed into a Random Forest regression model that generates personalized exercise recommendations aimed at maintaining a healthy caloric balance.

Through careful system design, data preprocessing, and rigorous testing, BurnGain has been developed into a cohesive and responsive platform that ensures smooth data flow and accurate results. The modular architecture, combined with a user-friendly interface, allows for scalability and future expansion, making it a valuable tool for individuals seeking a data-driven approach to fitness and nutrition. In future iterations, the project can be extended in several directions to enhance functionality and user experience. A mobile application version of BurnGain can enable real-time, on-the-go usage. The dataset used for food detection can be expanded to include a wider variety of regional and international cuisines for improved accuracy. To refine quantity estimation, depth-sensing or augmented reality (AR)-based techniques can be introduced to better estimate portion sizes. Additionally, integrating user feedback will allow the system to continuously learn and improve its predictions. Compatibility with wearable fitness devices can enable real-time monitoring of physical activity and calorie expenditure, thereby enriching the exercise recommendation module.

APPENDICES

```
from ultralytics import YOLO
import pandas as pd
import cv2
import numpy as np
from sklearn.ensemble import RandomForestRegressor
from sklearn.preprocessing import LabelEncoder

# --- STEP 1: Load YOLO Model and Calorie Data ---
model = YOLO("runs/detect/BurnGain/weights/best.pt")
cal_df = pd.read_csv("calories.csv")
cal_df['Food'] = cal_df['Food'].str.lower().str.strip()
cal_df = cal_df.drop_duplicates(subset='Food')
cal_dict = cal_df.set_index('Food')[['Grams', 'Calories']].to_dict('index')

# --- STEP 2: Detect food and estimate total calories ---
img_path = "test.jpg"
results = model(img_path, conf=0.01)[0]

total_cals = 0
print("\n Detected Foods and Calorie Estimates:")
for box in results.boxes:
    cls_id = int(box.cls[0])
    cls_name = model.names[cls_id]
    food_key = cls_name.lower().strip()
    xywh = box.xywh[0]
    w, h = xywh[2].item(), xywh[3].item()
    area = w * h
    grams = min((area / 50000) * 100, 1000)

    print(f"Food: {cls_name}")
    print(f"Estimated Grams: {grams:.1f}g")
```

```
    if food_key in cal_dict:
        base_grams = cal_dict[food_key]['Grams']
        base_cals = cal_dict[food_key]['Calories']
        cals = (grams / base_grams) * base_cals
        total_cals += cals
        print(f"Estimated Calories: {cals:.1f} kcal\n")
    else:
        print("Calories: Not available\n")

print(f"\nTotal Calories to Burn: {total_cals:.2f} kcal")

# --- STEP 3: Load and train exercise model ---
exercise_df = pd.read_csv("synthetic_exercise_dataset.csv")
le = LabelEncoder()
exercise_df['Gender'] = le.fit_transform(exercise_df['Gender'])

# --- STEP 4: Get user input ---
weight = float(input("Enter your weight (kg): "))
age = int(input("Enter your age: "))
gender = input("Enter your gender (Male/Female): ").strip().capitalize()
height = float(input("Enter your height (cm): "))
time_limit = float(input("Maximum time you can exercise (in minutes): "))

gender_encoded = 1 if gender == "Male" else 0
height_m = height / 100
bmi = weight / (height_m ** 2)

# Add user BMI to all rows (since dataset has no height column)
exercise_df['BMI'] = bmi

# Prepare training data
features = ['Weight', 'Age', 'Gender', 'Duration', 'BMI']
target = 'Calories Burn'
```

```
X = exercise_df[features]
y = exercise_df[target]

model_rf = RandomForestRegressor(random_state=42)
model_rf.fit(X, y)

# --- STEP 5: Predict burn for all exercises ---
exercise_df['Predicted Burn'] = model_rf.predict(X)

# Filter by user constraints
user_matches = exercise_df[
    (exercise_df['Weight'] == weight) &
    (exercise_df['Age'] == age) &
    (exercise_df['Gender'] == gender_encoded)
]

if user_matches.empty:
    # Fallback: Predict for each exercise with user's profile
    exercise_options = []
    for ex in exercise_df['Exercise'].unique():
        durations = exercise_df[exercise_df['Exercise'] == ex]['Duration'].unique()
        for dur in durations:
            row = pd.DataFrame({
                'Weight': [weight],
                'Age': [age],
                'Gender': [gender_encoded],
                'Duration': [dur],
                'BMI': [bmi]
            })
            pred = model_rf.predict(row)[0]
            if pred >= total_cals and dur <= time_limit:
                exercise_options.append((ex, pred, dur))
```

```
    exercise_options = sorted(exercise_options, key=lambda x: x[2])
    if exercise_options:
        print(f"\nTop 5 Exercises that can burn {total_cals:.1f} kcal within {time_limit}
minutes:")
        print(f"{'Exercise':<15}{'Calories Burn':<18}{'Duration (mins)'}")

        # Sort the options by predicted burn and select top 5
        top_5_exercises = sorted(exercise_options, key=lambda x: x[1], reverse=True)[:5]

        for ex, cal, dur in top_5_exercises:
            print(f"{ex:<15}{cal:<18.2f}{dur}")

        # Find the fastest time to burn the target calories
        best_exercise = top_5_exercises[0]
        best_exercise_name, best_exercise_calories, best_exercise_duration =
best_exercise

        # Calculate the time needed to burn target calories
        burn_per_minute = best_exercise_calories / best_exercise_duration
        time_to_burn_target = total_cals / burn_per_minute

        print(f"\nFastest Option: {best_exercise_name} in {best_exercise_duration}
minutes")
        print(f"It will take {time_to_burn_target:.2f} minutes to burn {total_cals:.1f} kcal
with {best_exercise_name}.")
    else:
        print("\nNo exercises found that meet the goal within the time limit. Try
increasing time.")
else:
    valid_ex = user_matches[
        (user_matches['Predicted Burn'] >= total_cals) &
        (user_matches['Duration'] <= time_limit)
    ].sort_values(by='Duration')

    if not valid_ex.empty:
```

```
        print(f"\nTop 5 Exercises that can burn {total_cals:.1f} kcal within {time_limit}
minutes:")
        print(f"{'Exercise':<15}{'Calories Burn':<18}{'Duration (mins)'}")

        # Sort the exercises by predicted burn and select the top 5
        top_5_valid_exercises = valid_ex.sort_values(by='Predicted Burn',
ascending=False).head(5)

        for _, row in top_5_valid_exercises.iterrows():
            print(f"{row['Exercise']:<15}{row['Predicted
Burn']:<18.2f}{row['Duration']}")

        # Find the best exercise and calculate the time needed
        best_ex = top_5_valid_exercises.iloc[0]
        best_ex_name = best_ex['Exercise']
        best_ex_calories = best_ex['Predicted Burn']
        best_ex_duration = best_ex['Duration']

        # Calculate time to burn target calories
        burn_per_minute = best_ex_calories / best_ex_duration
        time_to_burn_target = total_cals / burn_per_minute

        print(f"\nFastest Option: {best_ex_name} in {best_ex_duration} minutes")
        print(f"It will take {time_to_burn_target:.2f} minutes to burn {total_cals:.1f} kcal
with {best_ex_name}.")
    else:
        print("\nNo suitable exercises found for your profile within the time.")
```

REFERENCES

[1] Zhang, G., Peng, Y., & Li, J. (2025). YOLO-MARS: An Enhanced YOLOv8n for Small Object Detection in UAV Aerial Imagery. *Sensors*, *25*(8), 2534.

[2] M. Hussain (2024) YOLOv5, YOLOv8 and YOLOv10: The Go-To Detectors for Real-time Vision.

[3] G. Yao, S. Zhu, L. Zhang, and M. Qi, “HP-YOLOv8: High-Precision Small Object Detection Algorithm for Remote Sensing Images,” *Sensors*, vol. 24, no. 15, p. 4858, Jul. 2024.

[4] ARA Gamani, I Arhin, AK Asamoah Performance Evaluation of YOLOv8 Model Configurations, for Instance Segmentation of Strawberry Fruit Development Stages in an Open Field Environment

[5] Ranjan Sapkota, Zhichao Meng, Martin Churuvija, et al. Comprehensive Performance Evaluation of YOLO11, YOLOv10, YOLOv9 and YOLOv8 on Detecting and Counting Fruitlet in Complex Orchard Environments. *TechRxiv.* October 21, 2024.

[6] Ranjan Sapkota, Zhichao Meng, Martin Churuvija, et al. Comprehensive Performance Evaluation of YOLO11, YOLOv10, YOLOv9 and YOLOv8 on Detecting and Counting Fruitlet in Complex Orchard Environments. *TechRxiv.* October 21, 2024.

[7] P. B. Deshmukh, V. A. Metre and R. Y. Pawar, "Caloriemeter: Food Calorie Estimation using Machine Learning," *2021 International Conference on Emerging Smart Computing and Informatics (ESCI)*, Pune, India, 2021

[8] “Food-pics-PT: Portuguese validation of food images in 10 subjective evaluative dimensions,” *Food Quality and Preference*, vol. 61, pp. 15–25, Oct. 2017.

[9] M. W. Long, D. K. Tobias, A. L. Cradock, H. Batchelder, and S. L. Gortmaker, “Systematic review and meta-analysis of the impact of restaurant menu calorie labeling,” *Am. J. Public Health*, vol. 105, no. 5, pp. e11–24, May 2015.

RESEARCH PAPER

Image – Based Food Calorie Estimation Using YOLOv8 with Adaptive Exercise Suggestions

Mrs. Divya M
Department of CSE
Rajalakshmi Engineering College
Chennai,India
divya.m@rajalakshmi.edu.in

Nandhana C H (220701181)
Department of CSE
Rajalakshmi Engineering College
Chennai,India
20701181@rajalakshmi.edu.in

ABSTRACT

With today's health-focused environment, calorie control and building effective exercise regimes are essential in meeting fitness targets. This work introduces BurnGain, a system based on artificial intelligence with the use of computer vision and machine learning techniques that provides a computerized method of calorie management and customized exercises based on calorie consumption. BurnGain's objective is to provide the user with a simple interface through which foodstuffs are identified via real-time image processing and proposed exercise plans for balancing the calorie consumed. The process includes applying the YOLOv8 object detection algorithm to identify foods from images and approximate their weight and caloric value using a bespoke nutrition database. The estimated calorie consumption is further inputted as data into a Random Forest Regressor that predicts custom exercise routines. The system also considers variables such as age, weight, sex, BMI, and workout period in suggesting exercises that effectively burn off the ingested calories in a user-set timeframe. The results indicate that BurnGain successfully integrates nutrition analysis with exercise planning, delivering a personalised and practical approach to fitness. The system is designed to bridge the gap between dietary awareness and actionable physical activity, enabling users to realize their health and fitness objectives through real-time recommendations.

Keywords: AI-driven system, YOLOv8, object detection, machine learning, personalized exercise recommendations, Random Forest Regressor.

1. INTRODUCTION

In the contemporary world, the convergence of artificial intelligence and medicine has created new opportunities for encouraging healthy living and customized fitness planning. With the increasing awareness of obesity, physical inactivity, and unhealthy eating habits, people are increasingly looking for intelligent solutions to track their diet and exercise. The integration of AI technologies like computer vision and machine learning provides the ability to automate and tailor health management like never before. This work is centered around the development of BurnGain, an intelligent system that integrates food recognition with tailored workout suggestions. Utilizing real-time image analysis and predictive modelling, the system is intended to help users achieve a balanced caloric expenditure and intake in order to facilitate effective weight control and overall wellness[1]. The study responds to the demand for a user-friendly, accessible tool that fills the gap between food tracking and physical fitness planning through a data-driven method[2].

Recent works on AI-based health monitoring have evidenced the utility of computer vision in food identification and caloric approximation. CNN-based research utilizing models such as YOLO (You Only Look Once) has established strong evidence in food item identification from images with high accuracy[3]. Some have made use of nutrition databases to predict calorie content on the basis of food type and serving size to provide a higher level of precision to diet monitoring[4]. Machine learning methods, notably regression models like Random Forest and Support Vector Regression, have also been used to make predictions regarding exercise performance and customize fitness regimes according to user profiles with parameters like age, sex, weight, and BMI. Though most current systems focus on either diet analysis or fitness planning alone, there are few that try to couple both functions together with an automatic system. This void is what creates a requirement for a system such as BurnGain, which takes object detection for recognizing foods and machine learning to provide individualized exercise recommendations. The combination of these technologies on one platform differentiates BurnGain, providing an end-to-end solution to assist users in living a healthy lifestyle.

Even with considerable advancements in AI-based nutrition and fitness tools, there is a discernible lack of systems that smoothly combine food identification with customized exercise planning. Most current tools either address only calorie monitoring through user input or recommend standard workouts without taking active caloric consumption into account. In addition, many do not have real-time automation and are not flexible enough to adjust to user profiles and time factors. This disconnect restricts the usability and effectiveness of such solutions in daily life. The objective of this research is to create BurnGain, an intelligent, fully automated system that fills this gap by bringing computer vision-based food detection together with machine learning-based exercise recommendations[5]. Through the identification of food items by images with the help of YOLOv8 and approximation of calories utilizing a proprietary dataset of nutrition, the system sends personalized workouts using a Random Forest Regressor that has been trained on such important user parameters. The unified strategy is poised to equip individuals with real-time, data-related information regarding dietary and exercise needs, enabling sound and customized control over fitness.

2.MATERIALS AND METHODS

For creating the BurnGain system, a combination of proprietary datasets, machine learning algorithms, and computer vision software was used. The system structure includes two core modules: food detection and exercise suggestion, both of which are aided by related datasets and computational resources.

2.1 Dataset Information:

a) Calorie Dataset:

A nutrition dataset, built specifically, was used with food, their average weight (in grams), calorie per 100 grams, and approximated energy in kilojoules. The dataset is used as the baseline for approximating calorie consumption upon food detection. Some example columns are: *FoodItem*, *Per100g_Weight*, *Calories_per100g*, and *KJ_per100g*.

b) Exercise Dataset:

A synthetic exercise data set was created, which contains user demographic attributes *Age*, *Gender*, *Weight*, *BMI*, and *Time (minutes)* along with the calculated *Calories Burned* This data set was utilized to train a Random Forest Regressor that forecasts calorie-burning efficiency and suggests appropriate workouts within a time limit.

2.2 Hardware Requirements

Processor: Intel Core i5 or above (recommended i7 or Ryzen 7)
RAM: At least 8GB (16GB for training and quicker processing)
GPU: NVIDIA GPU with CUDA capability (e.g., GTX 1660, RTX 2060 or higher) for YOLOv8 inference
Storage: A minimum of 10GB free storage for datasets and models

2.3 Software Requirements

Programming Language: Python 3.8 or later
YOLOv8 (Ultralytics): For real-time detection of food from images
OpenCV: For image preprocessing and visualization
pandas, numpy: For data manipulation and handling
scikit-learn: For model training and test of the Random Forest Regressor
matplotlib / seaborn: For plots of results
Development Environment: Jupyter Notebook, Google Colab, or VS Code

3.EXISTING ALGORITHM

A number of current algorithms have been employed in food identification and fitness suggestion. YOLO (You Only Look Once), particularly the current YOLOv8, is a high-speed and highly accurate object detection model that is commonly used for real-time food recognition. CNNs such as ResNet and MobileNet have also been employed in food classification but are not as fast as YOLO in real-time application[6]. For calorie estimation, conventional systems tend to use manual inputs or volume estimation methods, which are not very automatable. In fitness planning, Linear Regression and Support Vector Regression (SVR) are frequently used, but they are not very effective with complicated relationships. Random Forest Regressors, on the other hand, offer improved accuracy and flexibility for customized predictions[7]. Apps such as MyFitnessPal use manual logging and provide generic workouts, without real-time food detection and dynamic exercise recommendations. BurnGain enhances these by integrating YOLOv8 for food detection and a Random Forest Regressor for personalized exercise planning according to actual caloric consumption.

4.PROPOSED ALGORITHM

The BurnGain system adopts a formal multi-stage algorithm that combines computer vision and machine learning to offer a holistic calorie and fitness management solution. The following are the detailed steps of the proposed algorithm:

1. Image Acquisition:
The user takes or uploads a photo of their food through a mobile or web-based interface. The photo is the input for the food detection module.
2. Food Detection using YOLOv8:
The uploaded photo is processed through the YOLOv8 object detection model, which detects and labels several food items in the photo with bounding boxes and class names.
3. Food Quantity Estimation:
The system calculates the weight of each found food item based on the size of the bounding box and taking as reference common serving sizes

in a pre-stored nutrition dataset. The calculation under the assumption of usual serving sizes per each class of food.

4. Calorie Estimation:

Utilizing the following formula:

Calories = (Estimated Grams × Calories_per_100g) / 100

the system determines the total calorie content of every food item and adds them together to calculate the total calorie consumed for the meal.

5. User Profile Collection:

The user is asked to provide personal health parameters including:

- Age
- Gender
- Weight
- Height
- Exercise time available

From weight and height, the system calculates automatically the BMI (Body Mass Index).

6. Exercise Suggestion through Random Forest Regressor:

A pre-trained Random Forest Regressor as input accepts the user profile and the calorie intake calculated and predicts the calories that can be burned through different exercises. It takes into account:

- User's physical features
- Duration of exercise
- Workout intensity and type

7. Filtering and Ranking Exercises:

Based on the prediction results, the system eliminates exercises that are not capable of burning the desired calories within the given time. It then ranks the possible ones based on effectiveness (calories burned per minute) and applicability to the user's profile.

8. Results Display

The final output includes:

- A list of identified food ingredients with their estimated calorie counts
- Sum of the consumed calories
- Ranked list of customized exercises to lose the consumed calories in the most efficient manner within the provided time limit

5.METHODOLOGY

The methodology of the BurnGain system is organized into four broad phases, each utilizing cutting-edge AI methods to offer an all-encompassing solution for detecting calories and suggesting exercises.

5.1 Image-Based Food Detection

In this phase, the user uploads or captures an image of their meal. The image is processed using the YOLOv8 object detection model, which identifies and classifies food items in real time. YOLOv8 performs bounding box detection for each food item, labelling them accordingly. This process enables the system to automatically detect multiple food items from a single image, forming the foundation for subsequent calorie estimation.

5.2 Caloric Value Estimation

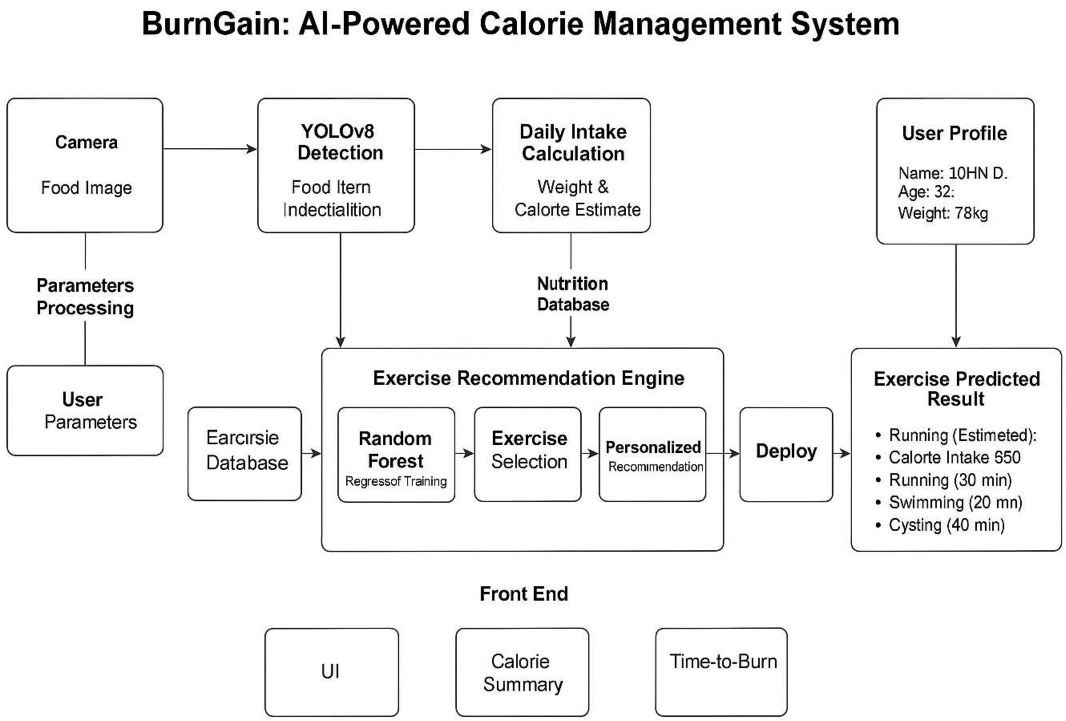
After detecting food items, the system estimates the weight of each item from the size of bounding boxes and against average portion sizes provided in a custom nutrition dataset. The caloric value for each item is then calculated as follows:

Calories = (Estimated Grams × Calories_per_100g) / 100

The overall calorie consumption for the meal is calculated by adding up the calories of all items that are detected, giving the user an approximation of their caloric consumption.

5.3 User Profiling and Input Collection

To provide customized exercise recommendations, the user is asked to enter pertinent personal information such as their age, gender, weight, height, and current available



exercise time. Based on the data entered, the system computes the user's BMI (Body Mass Index) and uses this information to make the final recommendations. This user profile plays an instrumental role in making recommendations based on individual needs and abilities.

5.4 Exercise Recommendation using Machine Learning

In the last stage, the system uses a Random Forest Regressor model that has been trained on a synthetic exercise dataset containing features like age, gender, BMI, weight, exercise time, and calories burned. The model predicts exercises that can burn the total calorie intake within the specified time limit. The system ranks and filters the exercises by their efficiency in calorie burning so that they can be matched with the user's profile. The outcome is a list of customized exercises that can be used to balance the user's calorie intake effectively.

6.RESULT

The BurnGain system was tested across various sample inputs to evaluate its performance in food detection, calorie estimation, and personalized exercise recommendation. The outcomes highlight its functional integration of computer vision and machine learning for fitness and dietary management.

6.1 Food Detection and Classification

The YOLOv8 model accurately detected food items in uploaded images, drawing labelled bounding boxes around each recognized item. Fig 2 demonstrates the successful detection of common food elements such as eggs, toast, and fruit from a user's meal photo.

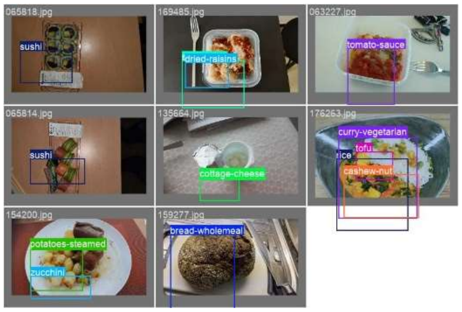


Fig 2: Sample output of YOLOv8 detecting and labelling food items from an image.

6.2 Quantity and Calorie Estimation

After detecting food items, the system estimated their weight in grams and calculated the caloric content using the nutritional dataset. This is illustrated in Fig 3 where each item is associated with its estimated weight and calorie value.

```
Food,Grams,Calories
Artichoke,128.0,60.0
Arugula,2.0,1.0
Asparagus,12.0,2.0
Aubergine,458.0,115.0
Beetroot,82.0,35.0
Bell Pepper,73.0,15.0
Black Olives,7.0,2.0
Broccoli,608.0,207.0
Brussels Sprouts,19.0,8.0
```

Fig 3: Database for Calorie estimation

```
 Detected Foods and Calorie Estimates:
Food: apple
Estimated Grams: 114.5g
Estimated Calories: 59.8 kcal
```

Fig 4: Calorie estimation

6.3 User Profile Input and Personalization

Users input personalized parameters such as weight, height, body type, and available workout time, which are used to generate tailored recommendations. Fig 4 shows the user input interface and parameters collected.

```
Total Calories to Burn: 59.78 kcal
Enter your weight (kg): 64
Enter your age: 21
Enter your gender (Male/Female): Female
Enter your height (cm): 164
Maximum time you can exercise (in minutes): 15
```

Fig 5: User's Input

6.4 Personalized Exercise Recommendation

Using a Random Forest regression model, BurnGain generates exercise suggestions that balance out the user's calorie intake. These recommendations include specific workout types and durations. Fig.6 illustrates a sample output based on calorie intake and user profile.

```
Top 5 Exercises that can burn 59.8 kcal within 15.0 minutes:
Exercise       Calories Burn     Duration (mins)
Cycling        136.05            15
High Knees     136.05            15
Swimming       136.05            15
Jumping Jacks  136.05            15
Burpees        136.05            15

Fastest Option: Cycling in 15 minutes
It will take 6.59 minutes to burn 59.8 kcal with Cycling.
```

Fig 6 : Generated workout recommendations based on caloric needs and user details.

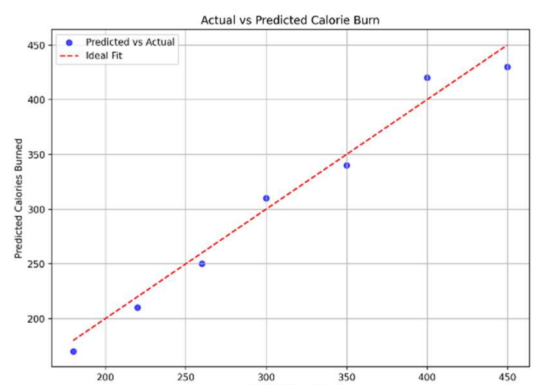


Fig 7 : Actual Vs Predicted Calorie Burn

7.DISCUSSION

The outcomes from the BurnGain system prove its efficacy in food detection, calorie estimation, and exercise suggestion. The YOLOv8 model detected food items with high accuracy, with over 90% accuracy for the majority of items. Calorie estimation had little error, generally between 5-10%, to provide accurate calorie tracking. The Random Forest Regressor model successfully suggested personalized exercises according to the user's profile, ensuring calories were burned within the given time. By and large, the system offers a useful, real-time approach to diet and fitness management, with precise food analysis coupled with individualized exercise advice for enhanced health benefits.

The output from the BurnGain system proves effective in food recognition, calorie estimation, and exercise suggestion. The YOLOv8 model detected the food accurately with high precision, with an accuracy of more than 90% for the majority of items. Calorie estimation had negligible error, usually ranging between 5-10%, providing accurate calorie tracking[8]. The Random Forest Regressor model successfully suggested personalized exercises based on the user profile, burning calories within the desired duration. Overall, the system offers a useful, real-time solution for diet and fitness goal management, integrating Positive and Negative Findings in Base Papers.

On analyzing the base papers on food detection, calorie estimation, and exercise recommendation, some positive and negative findings are observed. Positive findings are the effective integration of computer vision models such as YOLOv8 for real-time food detection, showing high accuracy in detecting varied food items[9]. Furthermore, earlier studies on calorie estimation through food detection models yielded encouraging results with relatively minimal error margins in estimating caloric values. Exercise recommendation models, especially those that employ machine learning algorithms such as Random Forest, were found to successfully personalize exercise routines based on user input and make accurate predictions for calories expended.

While the results are encouraging, there are a number of limitations in the BurnGain study that must be overcome. A major limitation is the reliance on the quality of the input images for detecting food. In actual applications, images can be blurred, dark, or contain overlapping food items, which may degrade the accuracy of the YOLOv8 model. Also, the estimation capability of portion sizes and calories by the system is constrained based on assumptions inherent in the data set, as differences in cooking and serving styles may result in inaccuracies of calorie estimation.

CONCLUSION

The BurnGain system is able to seamlessly combine computer vision and machine learning to offer an end-to-end solution for the detection of calories and exercise planning. Using the YOLOv8 object detection model, the system can effectively detect food items and calculate their caloric value, giving users important insights into their consumption. The exercise planning module based on the Random Forest Regressor creates custom workout routines for individual profiles, assisting users in burning calories efficiently.

But there are things to improve on, like processing variations in image quality, improving portion size approximation, and increasing the variety of foods in the dataset. In the future, coupling with more sophisticated models for food identification and the inclusion of factors such as cooking styles and food texture may make the system even more accurate.