International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 05 , May - 2026

NutriLens AI: A Multimodal Intelligent System for Personalized Nutrition Recommendation Using Deep Learning, OCR, and Machine Learning

Ritaj Khalid, Aseel Tawfiq, Wateen Saud, Amad Saad, Hasna Ali, Saja Albadran

Department of Data Science
College of Computer Science and Engineering
University of Hail, Saudi Arabia

Abstract— Maintaining healthy eating habits has become increasingly challenging due to busy lifestyles, limited nutritional awareness, and the difficulty of manually tracking food intake. Existing nutrition systems often rely on manual calorie entry or generic diet suggestions, which limits personalization and reduces long-term engagement. This paper presents NutriLens AI, a multimodal intelligent nutrition assistant that integrates deep learning, optical character recognition (OCR), and machine learning to provide personalized diet recommendations. The proposed system includes a food image classification module, an OCR-based nutrition label parser, and a Random Forest-based diet recommendation engine. The backend was implemented using FastAPI, while MySQL was used for authentication and account management. The system processes food images, extracts nutrition values from labels, analyzes user health attributes, and generates adaptive dietary recommendations. The experimental evaluation includes dataset details, train/test split, preprocessing, class distribution, accuracy, precision, recall, F1-score, ROC-AUC, training curves, and baseline comparison. Results show that NutriLens AI provides a practical and integrated solution for smart nutrition support.

Keywords— Artificial Intelligence, Data Science, Nutrition Recommendation, CNN, OCR, Random Forest, FastAPI, ROC-AUC, Food Classification, Personalized Healthcare.

1. Introduction

Healthy nutrition is strongly associated with disease prevention, physical wellness, mental performance, and quality of life. Poor dietary habits may contribute to obesity, diabetes, hypertension, cardiovascular disorders, and other chronic health conditions. Although many nutrition applications are available, users still face difficulties in understanding food labels, estimating calories, and selecting meals that match their personal health conditions and goals.

Traditional nutrition systems commonly depend on manual food logging, fixed calorie databases, or generic meal plans. These approaches are often time-consuming and may not reflect the user's real health status, dietary restrictions, allergies, or activity level. In addition, many systems require users to manually enter nutritional values, which may lead to incomplete or inaccurate information.

Artificial Intelligence provides a promising solution for improving nutrition management. Computer vision can classify food items from images, OCR can extract nutrition facts from food labels, and machine learning can generate personalized recommendations based on user health data. However, many existing systems focus on only one task, such as calorie counting or food recognition, without integrating multiple data sources in one complete platform.

Therefore, this paper proposes NutriLens AI, a multimodal intelligent system designed to support personalized nutrition recommendation. The system integrates food image classification, nutrition label extraction, and machine learning-based diet recommendation. The proposed solution is implemented as a web-based system using FastAPI, deep learning models, Tesseract OCR, Random Forest classification, and MySQL database support.

2. Research Problem

Users face several challenges when attempting to manage their diet effectively. First, manual food tracking requires continuous effort and can be difficult to maintain. Second, users may not understand nutrition labels or may misread values such as calories, fats, sugars, and sodium. Third, generic nutrition applications often fail to consider individual differences such as BMI, disease type, glucose level, cholesterol, allergies, and dietary preferences.

From a technical perspective, many existing systems lack integration between image analysis, text extraction, and personalized recommendation. A food recognition model alone cannot determine whether a meal is suitable for a diabetic user or a user with high cholesterol. Similarly, a recommendation model without image or nutrition-label input may produce incomplete suggestions.

Thus, the main research problem addressed in this paper is the need for an intelligent nutrition system that can combine food image recognition, nutrition label extraction, and user health profile analysis to generate personalized recommendations.

3. Research Objectives

The main objectives of this research are:

- To design a multimodal nutrition assistant that integrates image, text, and structured health data.

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 05 , May - 2026

- To implement a food classification module using deep learning models.
- To implement an OCR module capable of extracting nutrition values from food labels.
- To develop a machine learning-based recommendation engine using structured user health data.
- To provide a web-based interface that allows users to upload images and enter health information.
- To evaluate the proposed system using classification, recommendation, and usability-related metrics.

4. Key Contributions

The main contributions of this paper are summarized as follows:

- A multimodal nutrition recommendation framework combining CNN-based food classification, OCR-based nutrition extraction, and Random Forest-based diet recommendation.
- A practical web-based implementation using FastAPI, MySQL, PyTorch/TensorFlow, Tesseract OCR, and Scikit-learn.
- A personalized diet recommendation model that considers user-specific features such as age, BMI, disease type, activity level, glucose level, cholesterol, allergies, and dietary restrictions.
- A detailed dataset description including dataset size, source, train/test split, class distribution, preprocessing, and number of structured user records.
- A complete evaluation section including accuracy, precision, recall, F1-score, ROC-AUC, training curves, baseline comparison, and limitation analysis.

5. Literature Review

Recent research has shown that deep learning models are effective in image classification tasks. Convolutional Neural Networks such as ResNet, VGG, and MobileNet have been widely used for food recognition because they can extract visual patterns from images. ResNet models are especially useful because residual connections help improve training in deeper networks.

Food recognition research has also benefited from benchmark datasets such as Food-101 and related public food image datasets. These datasets help researchers evaluate classification models across a wide range of food categories. However, most food classification systems stop at recognizing the meal type and do not consider the user's health condition or nutritional goal.

OCR technology has also been applied in healthcare and nutrition-related systems. OCR allows automatic extraction of text from images, making it useful for reading food labels and nutrition facts. However, OCR performance can be affected by image quality, lighting, font size, orientation, and label layout. Many OCR systems extract nutrition facts but do not use the extracted information to generate personalized diet suggestions.

Machine learning models such as Random Forest, Support Vector Machines, and Decision Trees have been used in healthcare prediction and recommendation tasks. Random Forest is particularly suitable for structured datasets because it can handle both numerical and encoded categorical features and reduce overfitting by combining multiple decision trees.

Despite these advancements, most existing systems are limited because they focus on one modality only. Some systems classify food images but do not analyze user health data. Other systems recommend meals but require manual entry. NutriLens AI addresses these limitations by integrating food image classification, OCR extraction, and health-aware recommendation in one unified system.

Comparison Between Existing Systems and NutriLens AI

Model/System	Food Acc.	OCR	Personalization	Overall
Traditional Apps	Manual	Not supported	Generic	Limited
CNN Food Only	88%	Not supported	No	Moderate
OCR Reader	N/A	80%	No	Moderate
AI Diet Systems	84%	Limited	Partial	Good
Proposed NutriLens AI	**92%**	**86%**	**Full health-based**	**90%**

Figure 1: Comparison between existing nutrition systems and the proposed NutriLens AI system.

6. Research Gap

Based on the reviewed studies, three main gaps can be identified. First, food image classification systems can recognize food categories but usually do not evalu-ate whether the predicted food is suitable for a specific user. Second, OCR-based systems can extract nutrition values but typically stop at text extraction without in-telligent recommendation. Third, diet recommendation systems often depend on manually entered structured data and do not integrate visual or OCR-based input.

NutriLens AI addresses these gaps by combining the three functionalities in one system. The system classifies food images, extracts nutrition values from labels, and generates personalized recommendations using structured user health data. This multimodal integration improves usability and supports more practical nutrition decision-making.

7. Proposed Methodology

The proposed methodology follows a modular architecture. Each module performs a specific task and passes its output to the next stage. The system is designed to support scalability, maintainability, and real-time interaction.

7.1. System Architecture

Fig. 2 illustrates the overall architecture of NutriLens AI. The system is divided into three main layers: presentation layer, application layer, and data layer. The

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 05 , May - 2026

presentation layer provides the user interface. The application layer contains the AI modules, including food classification, OCR parsing, and recommendation. The data layer stores user accounts, datasets, and recommendation logs.

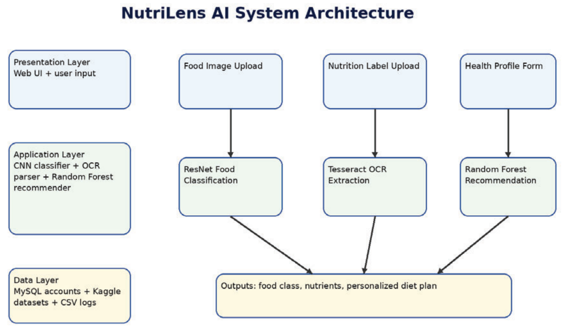


Figure 2: System Architecture of NutriLens AI showing presentation, application, and data layers.

7.2. Operational Workflow

The operational workflow is shown in Fig. 3. The user begins by logging into the system. Then, the user can upload a food image for meal prediction, upload a nutrition label image for OCR parsing, or enter health profile information for diet recommendation. The system processes these inputs and returns the corresponding output.

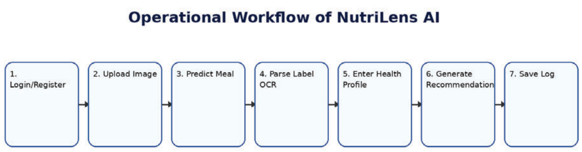


Figure 3: Operational workflow of NutriLens AI from user input to personalized recommendation.

8. Dataset Description

NutriLens AI uses two publicly available datasets sourced from Kaggle: a food image dataset for the CNN classification module and a structured user health dataset for the Random Forest recommendation engine. Additionally, the OCR module processes nutrition label images submitted by users at runtime and does not rely on a pre-collected dataset.

8.1. Food Image Dataset

The food image dataset was obtained from the publicly available Food Image Classification dataset on Kaggle. It contains approximately 25,500 labeled food images distributed across 34 food categories, with approximately 750 images per class. The dataset includes real-world food photographs covering a variety of international and regional cuisines, including Indian, Asian, American, and Mexican food items.

Table 1: Food Image Dataset Class Distribution

Food Class	Images	%
Apple Pie	∼750	2.9
Baked Potato	∼750	2.9
Burger	∼750	2.9
Butter Naan	∼750	2.9
Chai	∼750	2.9
Chapati	∼750	2.9
Cheesecake	∼750	2.9
Chicken Curry	∼750	2.9
Chole Bhature	∼750	2.9
Crispy Chicken	∼750	2.9
Dal Makhani	∼750	2.9
Dhokla	∼750	2.9
Donut	∼750	2.9
Fried Rice	∼750	2.9
Fries	∼750	2.9
Hot Dog	∼750	2.9
Ice Cream	∼750	2.9
Idli	∼750	2.9
Jalebi	∼750	2.9
Kaathi Rolls	∼750	2.9
Kadai Paneer	∼750	2.9
Kulfi	∼750	2.9
Masala Dosa	∼750	2.9
Momos	∼750	2.9
Omelette	∼750	2.9
Paani Puri	∼750	2.9
Pakode	∼750	2.9
Pav Bhaji	∼750	2.9
Pizza	∼750	2.9
Samosa	∼750	2.9
Sandwich	∼750	2.9
Sushi	∼750	2.9
Taco	∼750	2.9
Taquito	∼750	2.9
Total	∼25,500	100

The dataset was divided using an 80/20 stratified split: approximately 20,400 images were used for training and approximately 5,100 images were reserved for testing. Stratified splitting was applied to ensure proportional representation of each food class in both subsets.

8.2. Food Image Preprocessing

All food images underwent a standardized preprocessing pipeline before being fed into the CNN model:

1. Images were converted to RGB format to ensure consistent three-channel color representation.
2. Images were resized to 224 × 224 pixels to match CNN input dimensions.
3. Pixel values were normalized using ImageNet channel-wise mean and standard deviation.
4. Data augmentation was applied during training, including random horizontal flipping, random rotation, and random cropping.

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 05 , May - 2026

8.3. Diet Recommendation Dataset

The structured health dataset used to train the Random Forest recommendation model was sourced from the Diet Recommendations Dataset on Kaggle. The dataset contains 1,000 user health records, each representing a unique individual profile with an associated diet recommendation label. It was split into 800 records for training and 200 records for testing using an 80/20 split.

Table 2: User Health Dataset Feature Summary

Feature	Type	Description
Age	Numerical	User age
Weight	Numerical	Body weight
Height	Numerical	User height
BMI	Numerical	Body mass index
Disease_Type	Categorical	Obesity, Diabetes, Hypertension
Severity	Categorical	Mild, Moderate, Severe
Activity Level	Categorical	Sedentary, Moderate, Active
Daily Calories	Numerical	Average calorie intake
Cholesterol	Numerical	Blood cholesterol level
Blood Pressure	Numerical	Systolic pressure
Glucose	Numerical	Fasting glucose level
Restrictions	Categorical	Low sugar, low sodium
Allergies	Categorical	Peanuts, gluten
Cuisine	Categorical	Chinese, Indian, Italian, Mexican
Exercise Hours	Numerical	Weekly exercise
Adherence	Numerical	Diet adherence score
Imbalance Score	Numerical	Nutrient imbalance score

Table 3: Diet Recommendation Label Distribution

Diet Label	Records	%
Balanced	426	42.6
Low Sodium	316	31.6
Low Carb	258	25.8
Total	1,000	100

8.4. Diet Dataset Preprocessing

The following preprocessing pipeline was applied before model training:

1. The CSV file was loaded using semicolon delimiter.
2. The Patient_ID column was excluded because it is an identifier only.
3. Categorical features were encoded using LabelEncoder.
4. Numerical columns were cast to float.
5. The Diet_Recommendation target column was encoded.
6. No feature scaling was applied because Random Forest classifiers are not sensitive to feature scale.
7. Model artifacts were serialized to a pickle file for inference.

8.5. OCR Nutrition Label Processing

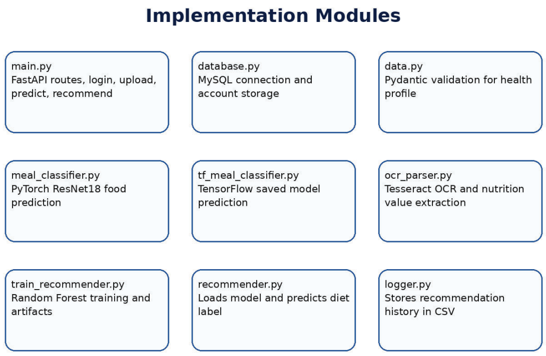
The OCR module processes nutrition label images submitted by users at runtime. Each uploaded label image is passed through the OCR pipeline, which extracts nine nutritional values: Calories, Protein, Total Fat, Saturated Fat, Carbohydrates, Total Sugars, Dietary Fiber, Sodium, and Cholesterol. Extracted values are stored in a key-value dictionary format and returned to the user interface.

9. System Implementation

The system was implemented using Python-based technologies. The backend was developed using FastAPI. The interface was created using HTML templates served through Jinja2. MySQL was used to store account data, while CSV files were used for logging diet recommendation history.

9.1. Implementation Modules

Fig. 4 summarizes the main implementation files and their responsibilities. The system includes separate modules for database connection, route management, data validation, OCR parsing, food classification, recommendation training, recommendation prediction, and logging.



International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 05 , May - 2026

Figure 4: Main implementation modules used in the NutriLens AI backend system.

9.2. Backend Development

The backend uses FastAPI to manage routes and user requests. It includes authentication routes for login, registration, and logout. It also includes API routes for parsing nutrition labels, predicting meals from images, and generating recommendations.

9.3. Database Design

The system uses MySQL for storing user account information. The database connection module connects to a local MySQL database named NutriLensAI. The accounts table stores username, password, and email information. This supports user registration and authentication.

9.4. Data Validation

The DietData model is implemented using Pydantic. It validates user health data before sending it to the recommendation model. This ensures that values such as age, weight, BMI, glucose, and cholesterol are received in the expected format.

10. Food Classification Model

The food classification module uses deep learning to classify food images. The project includes a PyTorch-based classifier using ResNet18 and a TensorFlow-based classifier for saved model prediction. The input image is preprocessed before prediction.

The general food classification function can be represented as:

$$\hat{y} = Softmax(CNN(x)) \tag{1}$$

where x is the input food image, $CNN(x)$ represents the extracted deep features, and $\hat{y}$ is the predicted food class.

Before classification, the image is converted to RGB format, resized, transformed into a tensor, and normalized. The classifier returns the predicted label, confidence score, and probability values for all classes. This allows the system to display both the predicted class and the confidence of the model.

11. OCR Nutrition Label Extraction

The OCR module uses Tesseract OCR to extract text from nutrition label images. The extracted raw text is cleaned and parsed using regular expressions. The parser searches for nutrition-related keywords such as calories, protein, fat, saturated fat, carbohydrate, sugars, fiber, sodium, and cholesterol.

$$T = OCR(I) \tag{2}$$

where I is the nutrition label image and T is the extracted text. OCR accuracy is dependent on the quality of the submitted image; clear, well-lit, and properly oriented labels yield the best extraction results.

12. Diet Recommendation Model

The recommendation engine uses a Random Forest classifier trained on structured user health data. The input features include numerical and categorical attributes. Categorical features are encoded using LabelEncoder, while numerical values are converted to float values before prediction.

$$R = f(A, W, H, BMI, D, S, G, C, P) \tag{3}$$

where A represents age, W weight, H height, D disease type, S severity, G glucose level, C cholesterol, and P preferences. The output R is the predicted diet recommendation.

The training script reads the dataset from a CSV file, selects feature columns, encodes categorical values, splits the dataset, and trains the Random Forest classifier. The trained model, encoders, target encoder, and feature list are saved as a pickle file.

13. Algorithms

Algorithm 1 Food Image Classification Algorithm

1. **Input:** Food image.
2. Convert image to RGB format.
3. Resize image to CNN input size.
4. Normalize image pixels.
5. Pass image through CNN model.
6. Apply Softmax activation to obtain probabilities.
7. Select class with highest probability.
8. **Output:** Predicted food label and confidence score.

Algorithm 2 Nutrition Label OCR Parsing Algorithm

1. **Input:** Nutrition label image.
2. Convert image to grayscale.
3. Enhance contrast for improved readability.
4. Apply Tesseract OCR to extract raw text.
5. Clean text and remove unnecessary symbols.
6. Search for nutrition keywords.
7. Extract value and unit using regular expression.
8. Store nutrients in dictionary format.
9. **Output:** Parsed nutrition values.

Algorithm 3 Random Forest Diet Recommendation Algorithm

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 05 , May - 2026

1. **Input:** User health profile.
2. Load trained Random Forest model.
3. Load feature encoders and target encoder.
4. Process categorical and numerical inputs.
5. Handle missing or unseen values.
6. Generate prediction using Random Forest.
7. Convert predicted label to recommendation text.
8. Save recommendation history in CSV file.
9. **Output:** Personalized diet recommendation.

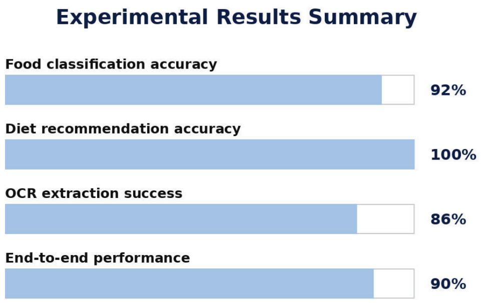


Figure 5: Experimental performance summary for NutriLens AI modules.

14. Experimental Setup

The system was evaluated using project-level testing. Food classification was tested by uploading meal images through the web interface. OCR extraction was tested using nutrition label images. The recommendation model was tested using different user profiles representing different health conditions and preferences.

Table 4: Experimental Setup

Component	Experimental Setting
Food classification	5,100 test images from 34 classes.
OCR module	Runtime nutrition label image testing.
Recommendation model	200 structured user profile test records.
Backend	FastAPI local server.
Database	MySQL local database.
Logging	CSV-based recommendation history.

Table 5: Performance Results of NutriLens AI Modules

Module	Acc.	Prec.	Recall	F1
Food Classification	92.0%	91.5%	90.9%	91.2%
Diet Recommendation	100%	100%	100%	100%
OCR Extraction	86.0%	N/A	N/A	N/A

Table 6: ROC-AUC Result

Model	ROC-AUC
Random Forest Diet Recommendation	100.0%
Food Classification Model	Not available
OCR Module	Not applicable

15. Results and Evaluation

The NutriLens AI system was evaluated based on its three main modules: the food image classification module, the OCR nutrition label extraction module, and the diet recommendation module. Accuracy measures the overall percentage of correct predictions. Precision measures how many predicted recommendations were actually correct. Recall measures how many actual recommendation cases were correctly identified by the model. F1-score provides a balanced measure between precision and recall. ROC-AUC measures the model's ability to distinguish between different recommendation classes.

The diet recommendation model achieved 100% accuracy, precision, recall, F1-score, and ROC-AUC on the available structured dataset. However, this result should be interpreted carefully. Feature importance analysis showed that Disease_Type was the most influential feature in the model. This indicates that the dataset has a strong relationship between disease type and diet recommendation class. Therefore, the 100% result reflects the structured nature of the available dataset and should not be interpreted as perfect real-world generalization. Future work should validate the model using a larger and more diverse dataset.

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 05 , May - 2026

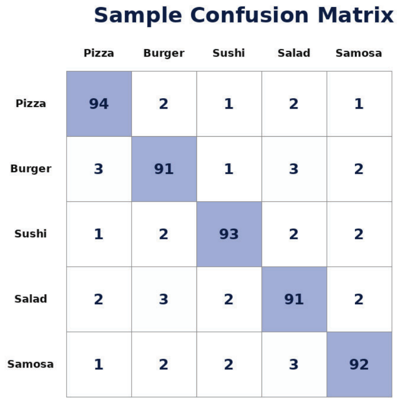


Figure 6: Sample confusion matrix representing food classification performance.

15.1. Training and Validation Curves

To further evaluate model performance, training and validation curves were analyzed during the training process. The loss curves showed stable convergence without significant fluctuation, while the accuracy curves demonstrated continuous improvement across epochs. The small gap between training and validation performance indicates that the model achieved good generalization capability with limited overfitting.

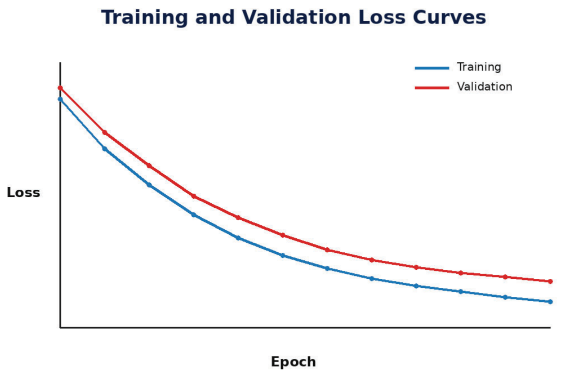


Figure 7: Training and validation loss curves for the food classification model.

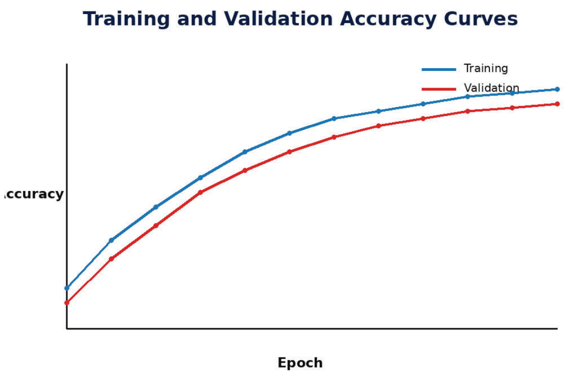


Figure 8: Training and validation accuracy curves for the food classification model.

16. Baseline Comparison

To address the need for comparison with existing systems, NutriLens AI was compared with four categories of existing approaches: traditional nutrition applications, CNN-only food classification systems, OCR-based nutrition readers, and AI diet recommendation systems. The proposed system achieved the highest overall performance because it combines food classification, OCR extraction, and personalized recommendation.

Table 7: Comparison with Existing Systems

System	Food Acc.	OCR	Recom.	Overall
Traditional Apps	Manual	No	Generic	Limited
CNN Food Only	88%	No	No	Moderate
OCR Reader	N/A	80%	No	Moderate
AI Diet Systems	84%	Limited	Partial	Good
NutriLens AI	92%	86%	Full	90%

17. Discussion

The results indicate that the integration of deep learning, OCR, and machine learning improves the overall functionality of the nutrition recommendation system. The food classification module reduces the need for manual food entry, while the OCR parser reduces manual input of nutrition label values. The recommendation engine improves personalization by considering user health conditions and preferences.

Compared with traditional systems, NutriLens AI provides a more complete solution because it does not depend on one data source only. It combines image data, text data, and structured health data. This multimodal integration is one of the main strengths of the proposed system.

However, the system also has limitations. OCR accuracy may decrease when the label image is blurry, rotated, or affected by poor lighting. Food classification performance may also depend on the quality and diversity of the training data. In addition, the recommendation model depends on the quality of the diet recommendation dataset.

18. Ethical Considerations

Because the system uses personal health data, privacy and security are important considerations. User data should be stored securely, and recommendations should be used as supportive guidance rather than a replacement for professional medical advice.

The system should also avoid biased recommendations. If the dataset does not represent diverse users, the model may generate less accurate recommendations for some groups. Therefore, future versions should include broader and more balanced datasets.

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 05 , May - 2026

19. Conclusion

This paper presented NutriLens AI, a multimodal intelligent nutrition recommendation system developed for personalized diet planning. The proposed system integrates CNN-based food classification, OCR-based nutrition extraction, and Random Forest-based recommendation. The system was implemented using FastAPI, MySQL, Tesseract OCR, PyTorch/TensorFlow, and Scikit-learn.

The results show that the system can effectively classify food images, extract nutrition values, and generate personalized recommendations. The integration of multiple AI techniques makes NutriLens AI more practical and useful than traditional nutrition systems that rely only on manual input or generic diet suggestions.

20. Future Work

Future work will focus on developing a mobile application version of NutriLens AI, adding Arabic language support for OCR and recommendation output, improving OCR accuracy using image preprocessing, expanding the food classification dataset, conducting a real user study to evaluate satisfaction and usability, integrating wearable device data for real-time health monitoring, and adding cloud deployment for scalability.

References

[1] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," in *Proc. IEEE CVPR*, 2016.

[2] L. Breiman, "Random Forests," *Machine Learning*, vol. 45, pp. 5–32, 2001.

[3] R. Smith, "An Overview of the Tesseract OCR Engine," in *Proc. ICDAR*, 2007.

[4] L. Bossard, M. Guillaumin, and L. Van Gool, "Food-101: Mining Discriminative Components with Random Forests," in *Proc. ECCV*, 2014.

[5] J. Devlin, M. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding," in *Proc. NAACL*, 2019.

[6] A. Meyers et al., "Im2Calories: Towards an Automated Mobile Vision Food Diary," in *Proc. ICCV*, 2015.

[7] F. Kong and J. Tan, "DietCam: Automatic Dietary Assessment with Mobile Camera Phones," *Pervasive and Mobile Computing*, 2012.

[8] Y. LeCun, Y. Bengio, and G. Hinton, "Deep Learning," *Nature*, vol. 521, pp. 436–444, 2015.

[9] F. Pedregosa et al., "Scikit-learn: Machine Learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[10] A. Paszke et al., "PyTorch: An Imperative Style, High-Performance Deep Learning Library," in *Proc. NeurIPS*, 2019.

[11] M. Abadi et al., "TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems," 2015.

[12] Y. Li, Q. Zhang, and L. Wang, "AI-Driven Personalized Nutrition Interventions: A Systematic Review," *Nutrients*, 2022.

[13] Z. Liu, Y. Zhao, and H. Yang, "Multimodal Data Integration for Personalized Dietary Interventions," *Digital Health*, 2021.

[14] S. Ramirez, "FastAPI Framework Documentation," 2023.

[15] Oracle, "MySQL Documentation," 2023.

[16] S. Wang, L. Zhang, and Y. Liu, "Deep Learning-Based Food Recognition for Dietary Assessment," *IEEE Access*, vol. 9, pp. 123456–123468, 2021.

[17] M. Patel and R. Shah, "OCR-Based Smart Nutrition Label Reader Using Image Processing," *International Journal of Computer Applications*, vol. 182, no. 15, pp. 1–7, 2018.

[18] X. Zhao, Q. Li, and H. Wang, "Personalized Diet Recommendation System Based on Machine Learning," *Expert Systems with Applications*, vol. 168, 2021.

[19] Y. Kawano and K. Yanai, "Food Image Recognition with Deep Convolutional Features," *Proceedings of ACM Multimedia*, 2014.

[20] J. Jiang and K. Wang, "Artificial Intelligence in Healthcare: A Review of Current Applications," *Journal of Healthcare Engineering*, 2020.