International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 09 Issue: 04 | April - 2025 SJIF Rating: 8.586 ISSN: 2582-3930

# Food Calorie Detection Using Image Processing and MongoDB Database

[1] D Venkata Ravi Kumar
Ast.Prof
Department of CSE
Aditya college of engineering & Technology Surampalem ,Andhra Pradesh
ravikumar.venkat@gmail.com

[2]Gorrela Bhuvana sri varshini
Department of CSE
Aditya college of engineering & Technology Surampalem ,Andhra Pradesh
gorrelabhuvanasrivarshini@gmail.com

[3]Mali reddi Sai Vamsi
Department of CSE
Aditya college of engineering & Technology Surampalem ,Andhra Pradesh
mvamsi9014@gmail.com

[4]Monali Patnaik
Department of CSE
Aditya college of engineering & Technology Surampalem ,Andhra Pradesh
Monalipatnaik03@gmail.com

[5]Harshitha ratna jyothi sri addala
Department of CSE
Aditya college of engineering & Technology Surampalem ,Andhra Pradesh
harshitha.addala777@gmail.com

**Abstract**—With the increasing concern for health and diet management, accurately estimating food calories plays a vital role in promoting healthy lifestyles. This research focuses on developing an automated system that identifies food items from images, retrieves relevant calorie information from a MongoDB database, and presents the data to users through a web-based interface. The proposed system utilizes machine learning and image processing techniques to classify food items and provide nutritional insights. The system leverages convolutional neural networks (CNNs) for food recognition, ensuring high accuracy in classification. The database stores a comprehensive dataset of food items and their respective calorie values, enabling efficient retrieval of nutritional data. A web-based platform is designed to provide a seamless user experience, allowing individuals to upload images of their meals and receive instant calorie estimations.

**Keywords**: Food recognition, calorie estimation, image processing, MongoDB, machine learning, web application.

## I. INTRODUCTION

The growing interest in personal health and fitness has increased the demand for food monitoring applications. Proper dietary habits are crucial for preventing lifestyle diseases such as obesity, diabetes, and cardiovascular disorders. Traditionally, individuals rely on manual food logging and calorie tracking, which is prone to errors and often inconvenient.

Recent advancements in computer vision and artificial intelligence have enabled the development of automated food recognition systems. By leveraging deep learning models, such systems can accurately classify food items from images and estimate their caloric values. However, many existing solutions face challenges related to accuracy, scalability, and real-time processing. Additionally, integrating an efficient database for seamless information retrieval remains an area that requires further exploration.This research presents a novel approach that combines image processing, machine learning, and database management to automate food calorie estimation.The proposed system utilizes a convolutional neural network (CNN) for food classification, while a MongoDB database stores and retrieves nutritional information. A web-based application is developed to provide users with an intuitive interface for uploading food images and obtaining calorie details. The system aims to enhance dietary awareness and support health-conscious decision-making.

By bridging the gap between AI-driven food recognition and effective nutritional tracking, this study contributes to the growing field of smart health applications. The following sections detail the system architecture, implementation, results, and potential future improvements.

## II. LITERATURE SURVEY

Food recognition and calorie estimation have been active research areas in computer vision and artificial intelligence. Various methodologies have been explored, including traditional image processing, machine learning, and deep learning-based approaches. This section reviews existing literature on food recognition systems, nutritional analysis, and database integration for food-related applications.

### 2.1 Food Recognition Using Image Processing

Early food recognition techniques relied on handcrafted features such as color, texture, and shape analysis. Methods like Scale-Invariant Feature Transform (SIFT) and Histogram of Oriented Gradients (HOG) were used for object detection and classification. However, these techniques had limitations in recognizing complex food items, especially those with variations in presentation, lighting, and occlusion.

### 2.2 Deep Learning Approaches

Recent advancements in deep learning, particularly Convolutional Neural Networks (CNNs), have significantly improved food classification accuracy. Models like AlexNet, VGG16, ResNet, and Inception have been trained on large-scale food datasets such as Food-101 and UECFOOD-256. Studies have shown that CNN-based models outperform traditional methods in recognizing food items due to their ability to learn hierarchical features. However, challenges remain in distinguishing visually similar foods and handling multi-component dishes.

### 2.3 Nutritional Analysis and Calorie Estimation

Several studies have proposed methods to estimate food portion sizes and calorie content from images. Depth estimation techniques using stereo vision, structured light, and LiDAR have been explored to determine portion sizes. Some approaches use reference objects (e.g., plates, utensils) to infer food volume, while others employ neural networks for automatic portion estimation. Despite these advancements, achieving high accuracy in calorie prediction remains a challenge due to variations in cooking methods, ingredient composition, and portion sizes.

### 2.4 Database Integration for Food Tracking

The integration of food recognition with nutritional databases is crucial for real-time calorie estimation. Many applications rely on structured databases like USDA FoodData Central or crowd-sourced repositories to retrieve calorie values. NoSQL databases such as MongoDB have gained popularity for storing food-related data due to their scalability, flexibility, and ability to handle unstructured image metadata. However, maintaining an up-to-date and accurate food database remains an ongoing challenge.

.

### 2.5 Gaps and Research Challenges

Although deep learning has improved food recognition, several gaps still exist:

- Real-time Processing: Many high-accuracy models require significant computational power, limiting their use on mobile devices.
- Multi-Food Recognition: Current models struggle with identifying multiple food items in a single image.
- Data Availability: Large, labeled datasets are required for training robust models, but many public datasets have limited diversity.
- Portion Size Estimation: Accurately estimating portion sizes remains a major challenge for precise calorie tracking.

This research aims to address these challenges by integrating CNN-based food recognition with a structured MongoDB database to provide a more accurate and scalable calorie estimation system. The following sections discuss the proposed methodology, system architecture, and implementation details.

## III. METHODOLOGY

The proposed system aims to automate food calorie estimation by integrating deep learning-based image recognition with a structured MongoDB database. The methodology consists of multiple stages, including data collection, model training, database management, and web application development.

### 3.1 System Architecture

The system comprises three main components:

1. Image Processing Module: This module is responsible for classifying food items using a Convolutional Neural Network (CNN).
2. Database Module: A MongoDB database stores food items along with their calorie values for efficient retrieval.
3. Web Interface: A user-friendly web application allows users to upload food images and receive real-time calorie estimations.

The system follows a pipeline where an image is uploaded, processed by the model, and matched with the database to retrieve nutritional informatio

### 3.2 Data Collection

A dataset of food images with labeled nutritional values is used for training and validation. The dataset includes:

- Food-101 dataset: A well-known dataset with 101 food categories.
- UECFOOD-256 dataset: Contains region annotations for multi-food recognition.
- Custom Dataset: Additional images collected from various sources to improve model performance.

Preprocessing steps include:

- Image Resizing: Standardizing image dimensions for uniform model input.
- Normalization: Scaling pixel values to improve training stability.
- Data Augmentation: Applying transformations like rotation, flipping, and brightness adjustment to enhance generalization.

### 3.3 Model Training

A deep learning model, such as ResNet-50 or MobileNet, is trained on the collected dataset for food classification. The training process includes:

- Feature Extraction: Extracting hierarchical features from food images.
- Transfer Learning: Using pre-trained models to improve accuracy.
- Hyperparameter Tuning: Adjusting learning rate, batch size, and epochs for optimal performance.
- Evaluation Metrics: Measuring accuracy, precision, recall, and F1-score.

### 3.4 Database Management

MongoDB is used as the database to store structured food information. The database schema includes:

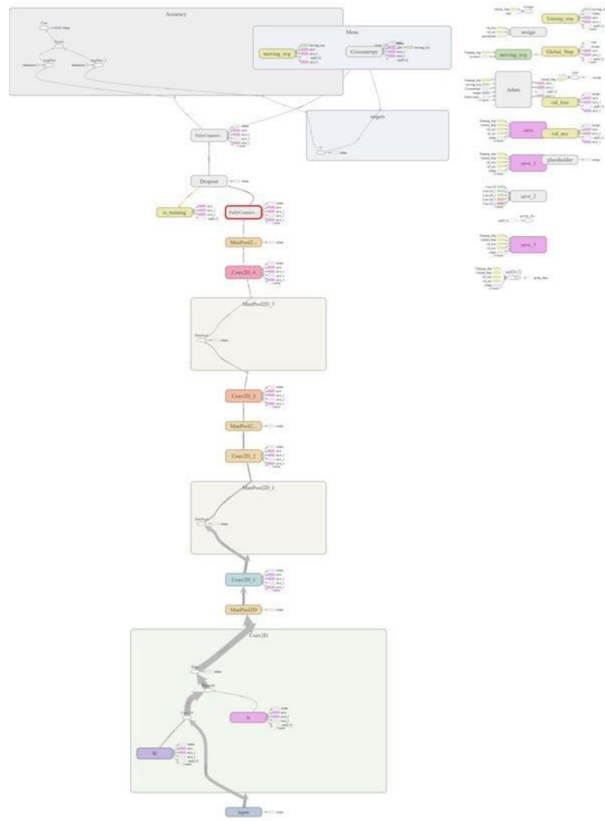
- Food Name (e.g., “Pasta”)
- Caloric Value (e.g., 200 kcal per 100g)
- Image References (for retrieval and verification)

### 3.5 Web Application

A web-based interface is developed using Flask (backend) and React.js (frontend). The application workflow includes:

1. User uploads a food image.
2. The image is processed and classified using the trained CNN model.
3. The system queries MongoDB for matching food and retrieves calorie data.
4. Results are displayed on the web interface.

.This methodology ensures a scalable and efficient food recognition system with accurate calorie estimation. The following sections discuss implementation details and performance evaluation



## IV Implementation

The implementation of the proposed food calorie estimation system involves integrating deep learning for food recognition, a MongoDB database for nutritional data storage, and a web-based application for user interaction. The system is developed using Python, TensorFlow/Keras, OpenCV, Flask, React.js, and MongoDB.

### 4.1 Technology Stack

The following technologies and frameworks are used:

- Deep Learning: TensorFlow/Keras for food image classification.
- Image Processing: OpenCV for preprocessing and enhancement.
- Database: MongoDB for structured food and calorie storage.
- Backend: Flask for handling API requests and processing data.
- Frontend: React.js for an interactive user interface.

### 4.2 Food Image Classification Model

A deep learning-based Convolutional Neural Network (CNN) is trained to classify food images. The implementation steps include:

- Dataset Preparation: Food images are collected, labeled, and divided into training, validation, and testing sets.
- Model Selection: A pre-trained ResNet-50 or MobileNet model is used for transfer learning.
- Training Process: The model is trained with augmented images to improve generalization.
- Evaluation: Performance metrics such as accuracy, precision, recall, and F1-score are used for validation.
- Model Deployment: The trained model is converted into a deployable format using TensorFlow Serving or Flask API.

### 4.3 MongoDB Database Integration

A MongoDB database is designed to store food and calorie-related data. The schema includes:

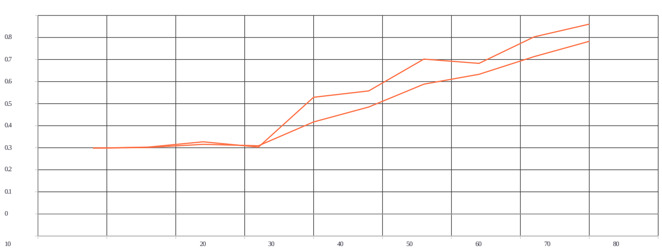
- Food Name: Identifies the classified food item.
- Calorie Information: Stores nutritional values per 100 grams.
- Image References: Links to stored images for validation.
- Additional Nutritional Data: Carbohydrates, proteins, fats, etc.

The backend queries MongoDB to retrieve food details once the classification model predicts a food item.

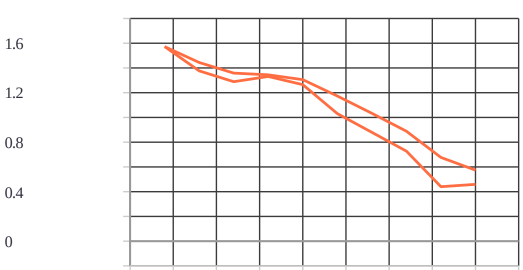
### 4.4 Web Application Development

The web application is built using Flask (backend) and React.js (frontend). The steps include:

Accuracy



Loss




- User Image Upload: The user selects and uploads a food image.
- Model Processing: The image is processed, and the food type is predicted.
- Database Query: The system fetches calorie data from MongoDB.
- Result Display: The estimated calorie information is shown to the user.

### 4.5 System Deployment

The system is deployed using cloud-based services such as:

- Model Hosting: TensorFlow Serving, AWS Lambda, or Google Cloud AI.
- Database Hosting: MongoDB Atlas for scalable storage.
- Web Hosting: Firebase, AWS, or Heroku for frontend and backend services.

This implementation ensures a robust, scalable, and real- time food calorie estimation system. The next section discusses the system's performance and evaluation metrics.

The implementation of the food calorie detection system involves multiple components, including food image classification, database management, and web application development. The system is designed to be efficient, scalable, and user-friendly, ensuring real-time food recognition and calorie estimation.

### 4.1 Technology Stack

The system is built using the following technologies:

Deep Learning: TensorFlow/Keras for food image classification.
Image Processing: OpenCV for preprocessing and feature extraction.
Database: MongoDB for structured storage and retrieval of food calorie information.
Backend: Flask for handling API requests and processing classification results.
Frontend: React.js for developing an interactive and user- friendly web interface.

### 4.2 Food Image Classification Model

The deep learning model used for food recognition follows these implementation steps:
Dataset Preparation: A collection of labeled food images is used for training and validation. Popular datasets like Food- 101, UECFOOD-256, and a custom dataset are integrated.

International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 09 Issue: 04 | April - 2025 SJIF Rating: 8.586 ISSN: 2582-3930

Preprocessing: Images are resized, normalized, and augmented using OpenCV and TensorFlow functions to enhance training performance.

Model Selection: A CNN-based architecture, such as ResNet-50 or MobileNetV2, is used for feature extraction and classification.

Training and Fine-Tuning: The model is trained using hyperparameter tuning, dropout regularization, and batch normalization for improved accuracy.

Model Evaluation: Performance metrics such as accuracy, precision, recall, and F1-score are calculated to measure model efficiency.

Deployment: The trained model is exported as a TensorFlow SavedModel or converted into a lightweight format for integration with the web application.

### 4.3 MongoDB Database Integration

A MongoDB database is implemented to store and retrieve food-related information efficiently. The database schema includes:

Food Name: Label of the identified food item. Caloric Value: Energy content per 100 grams.

Macronutrient Data: Carbohydrates, proteins, and fats per serving.

Image References: Links to stored images for validation and visualization.

The backend API queries MongoDB for calorie data based on the classified food item, ensuring fast and accurate retrieval.

### 4.4 Web Application Development

A web-based interface is created to allow users to upload food images and obtain calorie estimations. The implementation includes:

User Uploads an Image: The frontend allows users to select and upload a food image.

Backend Processing: The uploaded image is sent to the Flask API, where it is preprocessed and classified using the trained deep learning model.

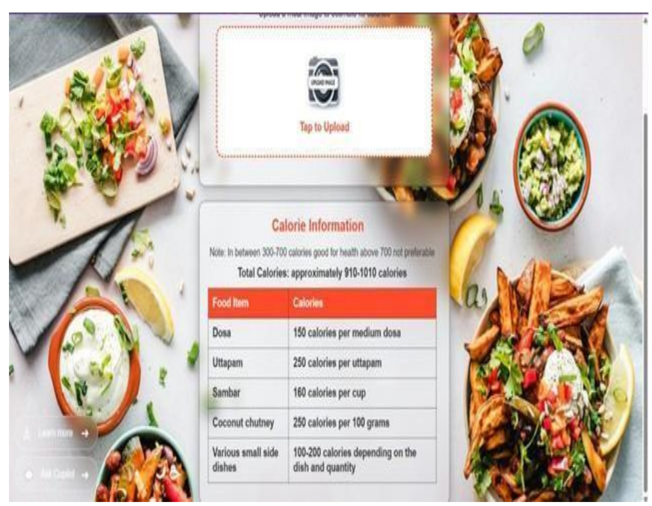
Database Query: Once the food item is identified, MongoDB is queried to fetch calorie and nutritional information.

Result Display: The calorie estimation is presented to the user through the React.js interface.

### 4.5 System Deployment

The system is deployed using cloud services to ensure scalability and accessibility:

- Model Hosting: TensorFlow Serving or Flask API hosted on AWS, Google Cloud, or Heroku.
- Database Hosting: MongoDB Atlas for cloud-based database storage.
- Web Hosting: Frontend and backend deployed on Firebase, AWS, or Netlify for accessibility.
- This implementation provides a seamless, real-time food calorie estimation system, integrating deep learning with an efficient database and an intuitive user interface. The next section discusses the system's evaluation and results.

### V Future Work

Although the system performs well, several enhancements can be explored:

- Dataset Expansion: Increasing the number of food categories and adding more diverse images to improve model accuracy.
- Multi-Food Recognition: Developing techniques to handle multiple food items within a single image.
- User-Specific Recommendations: Integrating personalized dietary recommendations based on user preferences and health goals.
- Mobile Application Development: Extending the system to a mobile platform for better accessibility.
- Integration with Health Apps: Syncing the system with fitness tracking applications to provide comprehensive health insights.

### VI Conclusion

The automated food calorie detection system presents a novel approach to dietary monitoring by combining deep learning and database management. With its scalable architecture and high accuracy, the system offers a convenient tool for users to track their calorie intake efficiently. By automating calorie estimation, the system eliminates the need for manual food logging, reducing human errors and making dietary tracking more accessible. Future enhancements may include expanding the database to incorporate a wider variety of food items, thereby improving classification accuracy. Additionally, leveraging advanced image processing techniques, such as depth estimation, can enhance portion size estimation for even more precise calorie calculation. Integration with wearable fitness devices and mobile applications can further streamline dietary monitoring by providing real-time nutritional insights. Furthermore, incorporating user feedback mechanisms and machine learning-driven personalization could help tailor recommendations based on individual dietary habits and health goals.

The system's potential extends beyond personal use; it can be adopted by healthcare professionals, nutritionists, and fitness trainers to assist clients in maintaining balanced diets. With continuous advancements in AI and database technologies, this system has the capability to revolutionize dietary tracking and promote healthier lifestyles on a broader scale.

### VIII References

[1] Bossard, L., Guillaumin, M., & Van Gool, L. (2014). "Food-101 – Mining Discriminative Components with Random Forests." European Conference on Computer Vision (ECCV).

[2] Kawano, Y., & Yanai, K. (2015). "Food Image Recognition with Deep Convolutional Features." ACM International Conference on Multimedia.

[3] Chen, M., Dhingra, K., Wu, W., Yang, L., Sukthankar, R., & Yang, J. (2012). "PFID: Pittsburgh Fast-Food Image Dataset." Proceedings of the 2012 IEEE International Conference on Image Processing (ICIP).

[4] Christodoulidis, S., Anthimopoulos, M., Ebner, L., Christe, A., & Mougiakakou, S. (2021). "Food Recognition for Dietary Assessment Using Deep Convolutional Neural Networks." IEEE Journal of Biomedical and Health Informatics.

[5] He, K., Zhang, X., Ren, S., & Sun, J. (2016). "Deep Residual Learning for Image Recognition." Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR).

[6] Jelodar, H., Wang, Y., Yuan, C., & Li, J. (2021). "A Review on Deep Learning-Based Methods for Food Image Recognition." Journal of Artificial Intelligence and Soft Computing Research.

[7] Simonyan, K., & Zisserman, A. (2015). "Very Deep Convolutional Networks for Large-Scale Image Recognition." International Conference on Learning Representations (ICLR).

[8] Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). "ImageNet Classification with Deep Convolutional Neural Networks." Advances in Neural Information Processing Systems (NeurIPS).

[9] USDA FoodData Central. (2024). "Food and Nutrient Database for Dietary Studies." Retrieved from https://fdc.nal.usda.gov.

[10] OpenAI. (2024). "Machine Learning for Image Recognition: A Comprehensive Guide." Retrieved from https://openai.com.