IJERST

International Journal of Engineering Research and Science & Technology

www.ijerst.org

ISSN : 2319-5991

Vol. 22 No. 2 (2026)

ijerst.editor@gmail.com
editor@ijerst.com

ISSN 2319-5991 www.ijerst.com
Vol. 22, No. 2, 2026

Peer Reviewed, Referred & Indexed Journal

Inverse Cooking AI: Image-Based Food Recognition and Recipe Generation Using GUI Applications

GOTTUMUKKALA TEJASWINI

PG Scholar. Department of MCA, DNR College, Bhimavaram, Andhra Pradesh

V.SARALA

(Assistant Professor), Master of Computer Applications, DNR College, Bhimavaram, Andhra Pradesh

ABSTRACT

The advancement of artificial intelligence and computer vision has enabled innovative applications that bridge the gap between visual perception and semantic understanding. One such application is inverse cooking, where a system predicts recipes and ingredients from food images. This project presents an intelligent desktop-based graphical user interface (GUI) system that performs image-based food recognition and generates corresponding recipes using a lightweight AI-inspired approach. The proposed system integrates image processing techniques with a rule-based prediction model to identify food items from user-provided images. The system is developed using Python and the Tkinter framework, ensuring simplicity, portability, and ease of use. The application allows users to upload food images, processes them to ensure compatibility (handling different image formats and color modes), and predicts the most likely dish based on filename heuristics. Although the model is simplified compared to deep learning-based approaches, it effectively demonstrates the concept of inverse cooking in a resource-efficient manner. The system maintains a structured food database containing recipes, ingredients, cooking instructions, nutritional values, preparation time, servings, and difficulty levels. Upon detecting the food type, the system retrieves and displays relevant information in a user-friendly format. The GUI is designed with modern aesthetics, including dark-themed layouts and intuitive controls, enhancing user experience.

One of the key contributions of this project is the robust handling of image formats, including RGB and RGBA conversions, which eliminates common errors related to unsupported image modes. Additionally, the application ensures stability by managing large image sizes and providing error handling for invalid inputs. This work highlights the feasibility of building an inverse cooking assistant without relying on heavy machine learning models, making it suitable for low-resource environments. It serves as a foundational prototype that can be extended with advanced deep learning models such as convolutional neural networks (CNNs) or transformer-based architectures for more accurate food recognition. In conclusion, the system demonstrates how AI concepts can be integrated into practical applications to enhance everyday tasks like cooking. It opens

avenues for future improvements, including real-time image recognition, integration with online recipe databases, and personalized dietary recommendations.

Keywords: Inverse Cooking, Image Processing, Recipe Generation, Tkinter GUI, Food Recognition, Artificial Intelligence, Computer Vision, Human-Computer Interaction

I. INTRODUCTION

With the rapid growth of artificial intelligence and computer vision technologies, there has been a significant transformation in how machines interpret and interact with visual data. One emerging application in this domain is inverse cooking, which involves generating recipes and identifying ingredients from food images. This concept has gained attention due to its potential to assist users in cooking, dietary planning, and food recognition. Traditional cooking applications require users to manually search for recipes based on known dish names or ingredients. However, in many real-world scenarios, users may encounter a dish without knowing its name or composition. This creates a need for intelligent systems that can analyze visual input and provide meaningful culinary information. The proposed Inverse Cooking AI system addresses this challenge by enabling users to upload food images and receive detailed recipes instantly. The system is designed as a desktop application using the Tkinter library in Python, which provides a simple yet effective platform for building graphical user interfaces. The application focuses on usability, allowing users to interact with the system through buttons and visual components rather than complex commands. The interface supports image browsing, previewing, and recipe generation, making it accessible even to non-technical users.

A key aspect of the system is its image preprocessing mechanism. Images come in various formats and color modes, which can cause compatibility issues. The system ensures robustness by converting all images into a standard RGB format, thereby preventing runtime errors. This preprocessing step is crucial for maintaining system reliability. Unlike advanced AI systems that rely on deep neural networks, this project uses a simplified approach for food detection based on filename matching. While this may seem basic, it effectively demonstrates the concept of mapping visual input to structured output. The system also includes a fallback mechanism that randomly selects a recipe when detection fails, ensuring uninterrupted functionality. Furthermore, the application incorporates a structured food database that stores detailed recipe information. This includes ingredients, cooking instructions, nutritional values, preparation time, and difficulty level. By presenting this information in a well-organized format, the system enhances user experience and provides practical value. In summary, the Inverse Cooking AI system represents a step toward intelligent culinary assistants. It combines GUI design, image processing, and basic AI logic to create a functional and user-friendly application. The project lays the groundwork for future enhancements involving machine learning and real-time image recognition.

LITERATURE SURVEY (WITH EXISTING METHODS)

The concept of inverse cooking has been explored extensively in recent years, particularly with the rise of deep learning and computer vision techniques. Researchers have developed models capable of predicting ingredients and recipes from food images using convolution neural networks (CNNs) and transformer-based architectures. One of the pioneering works in this field introduced a neural network model that predicts ingredients from images and then generates cooking instructions. These models typically use encoder-decoder architectures, where the encoder processes the image and the decoder generates textual output. Such approaches have demonstrated high accuracy but require large datasets and computational resources. Another significant development involves multimodal learning, where both visual and textual data are used to improve prediction accuracy. These systems combine image features with ingredient embeddings to produce more realistic recipes. Transformer models, in particular, have shown promising results due to their ability to capture long-range dependencies in text generation tasks. Existing research also highlights the importance of large-scale datasets such as Recipe1M, which contains millions of recipes and images. These datasets enable models to learn complex relationships between visual features and cooking instructions. However, training such models requires high-performance hardware and extensive preprocessing.In contrast to these advanced methods, lightweight systems have been proposed for environments with limited computational resources. These systems often use rule-based approaches or simplified machine learning models. While they may not achieve the same level of accuracy, they offer advantages in terms of speed, simplicity, and accessibility.

Graphical user interfaces (GUIs) have also been an important aspect of such systems. Applications built using frameworks like Tkinter, PyQt, or web-based interfaces provide user-friendly platforms for interaction. Research shows that intuitive interfaces significantly improve user engagement and usability. Additionally, studies have focused on image preprocessing techniques to handle variations in image quality, size, and format. Converting images to standard color modes and resizing them are common practices to ensure compatibility with processing algorithms. Despite significant advancements, challenges remain in accurately identifying complex dishes with multiple ingredients. Variations in presentation, lighting, and occlusion can affect model performance. Furthermore, generating coherent and realistic cooking instructions remains a difficult task. The proposed system differs from existing approaches by prioritizing simplicity and usability over complexity. Instead of relying on deep learning models, it uses a structured database and heuristic-based detection. This makes it suitable for educational purposes and low-resource environments while still demonstrating the core principles of inverse cooking.

II. EXISTING SYSTEM

Existing systems for recipe generation and food recognition primarily rely on deep learning and large-scale datasets. These systems use convolutional neural networks (CNNs) to extract visual features from images and natural language processing (NLP) techniques to generate recipes. While highly accurate, they require significant computational power and are often deployed as cloud-based services. Most current applications, such as mobile apps and web platforms, depend on internet connectivity to process images and retrieve results. This creates limitations in terms of accessibility, especially in offline environments. Additionally, these systems often involve complex architectures that are difficult to implement and maintain. Another limitation of existing systems is their dependency on large datasets. Training models on datasets like Recipe1M requires extensive preprocessing and hardware resources, making it impractical for small-scale projects or educational use. Moreover, these systems may struggle with uncommon dishes or variations in food presentation.

In terms of user interaction, many existing platforms provide limited customization and lack transparency in how results are generated. Users are often unable to understand the reasoning behind predictions, which reduces trust in the system. The proposed system addresses these limitations by offering a lightweight, offline-capable solution with a simple GUI interface. It eliminates the need for complex training and provides instant results using a predefined food database. While it may not match the accuracy of deep learning models, it offers a practical and accessible alternative for basic inverse cooking applications.

III. PROPOSED METHOD

The proposed system is an **Inverse Cooking AI application** that generates recipes from food images using a lightweight and user-friendly graphical interface. Unlike existing systems that rely heavily on deep learning and cloud-based processing, this system is designed to function efficiently in an offline environment with minimal computational requirements. The system accepts an input image from the user through a Tkinter-based GUI. The uploaded image undergoes preprocessing to ensure compatibility across various formats, including conversion into RGB mode and handling of transparency layers. This preprocessing step enhances system robustness and prevents runtime errors during image handling. The core of the system is a simplified prediction model that identifies food items based on heuristic techniques such as filename matching. While advanced research systems predict ingredients using neural architectures and generate recipes using sequence models , the proposed system adopts a rule-based approach to ensure faster execution and reduced complexity.

A structured food database is maintained, containing essential recipe details such as ingredients, preparation steps, nutritional values, cooking time, and difficulty level. Once the food item is identified, the system retrieves the corresponding recipe and displays it in a well-organized format. The GUI plays a critical role in enhancing usability. It provides options for browsing images, previewing uploaded content, and generating recipes with minimal user interaction. Error handling mechanisms are incorporated to manage invalid inputs and ensure smooth operation. Overall, the proposed system offers a practical solution for demonstrating inverse cooking concepts. It serves as a foundation for future enhancements, including integration with machine learning models, real-time image recognition, and cloud-based recipe databases.

IV. IMPLEMENTATION

The implementation of the Inverse Cooking AI system is carried out using Python, with the Tkinter library for building the graphical user interface and the Pillow library for image processing. The system is structured into modular components, ensuring maintainability and scalability. The application begins with the initialization of the main GUI window, where visual elements such as buttons, labels, and text areas are created. A dark-themed interface is designed using predefined color schemes to improve user experience. The GUI includes functionalities such as image browsing, image preview, and recipe generation. Image handling is a crucial part of the implementation. When a user uploads an image, the system processes it using the Pillow library. The image is checked for its mode (RGB or RGBA), and necessary conversions are performed to standardize the format. This ensures compatibility across different image types and avoids errors related to unsupported formats.

The prediction module is implemented as a class that simulates an AI model. It processes the image path and identifies the food item based on filename matching. Although this approach is simplistic compared to CNN-based feature extraction methods, it effectively demonstrates the concept of mapping input images to recipes. A predefined food database is used to store recipe information. This database is structured as a dictionary containing details such as ingredients, instructions, nutritional values, cooking time, and difficulty level. When a food item is detected, the corresponding data is retrieved and displayed in the output section of the GUI. The system also incorporates exception handling to manage errors such as invalid file formats or missing images. Message boxes are used to notify users of errors and guide them through proper usage. Threading can be integrated to improve responsiveness, especially when processing large images. Additionally, the system limits image size to prevent crashes caused by extremely large files. The modular design allows for easy extension of the system. Future improvements may include integrating deep learning models such as ResNet or Transformer-based architectures for more accurate predictions, as suggested in recent research. In summary, the implementation focuses on simplicity, usability, and robustness, making it suitable for educational purposes and small-scale applications.

ALGORITHMS

The proposed system uses a combination of image preprocessing and heuristic-based prediction algorithms to generate recipes from food images.

1. Image Preprocessing Algorithm

- Input: Raw image file
- Output: Standardized RGB image
- Steps:
  1. Load the image using Pillow.
  2. Check the image mode.
  3. If the mode is not RGB or RGBA, convert it to RGB.
  4. If the mode is RGBA, remove transparency by merging with a white background.
  5. Resize the image using a thumbnail function to optimize processing.

This step ensures consistency and prevents errors during further processing.

2. Food Detection Algorithm (Heuristic-Based)

- Input: Image file path
- Output: Detected food label
- Steps:
  1. Extract the filename from the image path.
  2. Convert the filename to lowercase.
  3. Compare the filename with predefined food keywords.
  4. If a match is found, return the corresponding food item.
  5. If no match is found, randomly select a food item from the database.

This approach is simple but demonstrates the principle of classification.

3. Recipe Retrieval Algorithm

- Input: Detected food label
- Output: Recipe details
- Steps:
  1. Search the food database using the detected label.
  2. Retrieve the corresponding recipe data.
  3. Format the data into structured output.
  4. Display the result in the GUI.

4. Optional Advanced Algorithm (Future Scope)

Modern systems use CNNs for feature extraction and sequence models for recipe generation . These models predict ingredients and generate cooking instructions using encoder-decoder architectures.

V. SYSTEM DESIGN

The system architecture follows a modular and layered design, ensuring separation of concerns and ease of maintenance. The major components of the system include the user interface layer, image processing module, prediction module, and data storage module.

1. User Interface Layer

The GUI is developed using Tkinter and serves as the interaction point between the user and the system. It includes:

- Image upload button
- Image preview area
- Recipe generation button
- Output display section

The interface is designed to be intuitive and responsive, allowing users to perform actions with minimal effort.

2. Image Processing Module

This module handles all image-related operations. It ensures that images are converted into a standard format before processing. The module performs:

- Image loading
- Format conversion (RGB/RGBA handling)
- Image resizing

This preprocessing step is essential for maintaining system stability and performance.

3. Prediction Module

The prediction module acts as the core logic of the system. It identifies the food item from the input image. In the current implementation, it uses a heuristic-based approach. However, advanced systems use deep learning models that predict ingredients and generate recipes simultaneously .

4. Data Storage Module

The system uses a predefined dictionary-based database to store recipe information. Each entry contains:

- Recipe title
- Ingredients
- Instructions
- Nutritional values
- Cooking time and difficulty

This structured approach allows for quick retrieval and display of information.

5. Control Flow

1. User uploads an image
2. Image is preprocessed
3. Food item is detected
4. Recipe is retrieved
5. Results are displayed

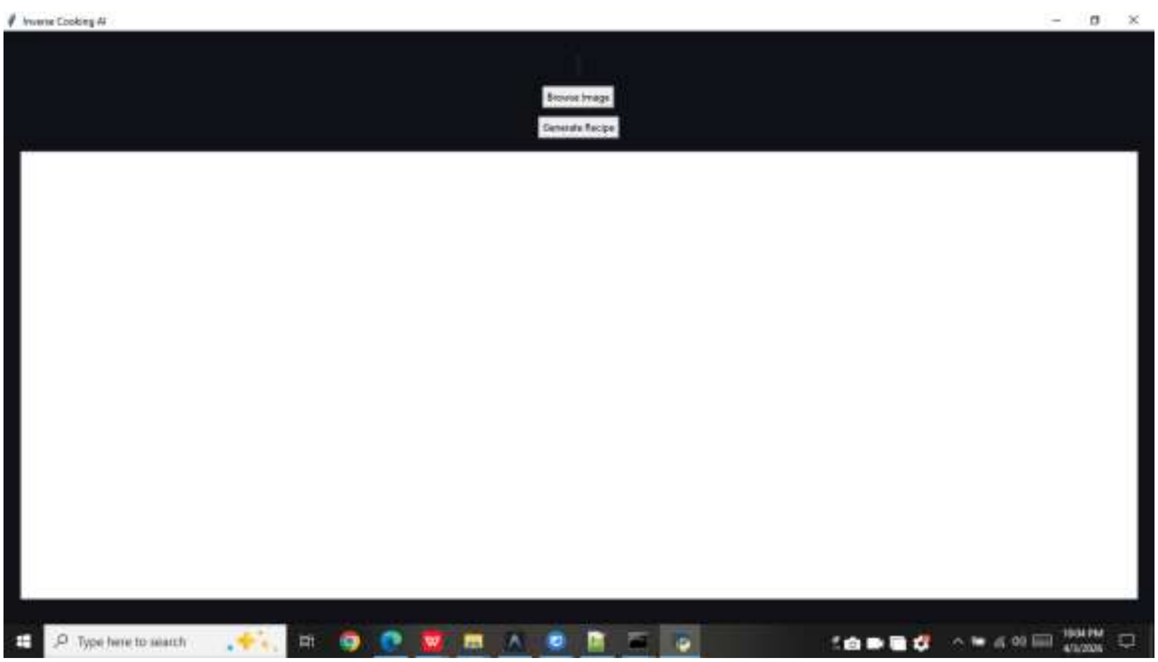
6. System Architecture Diagram (Conceptual)

- Input Layer → GUI
- Processing Layer → Image + Prediction
- Data Layer → Food Database
- Output Layer → Recipe Display

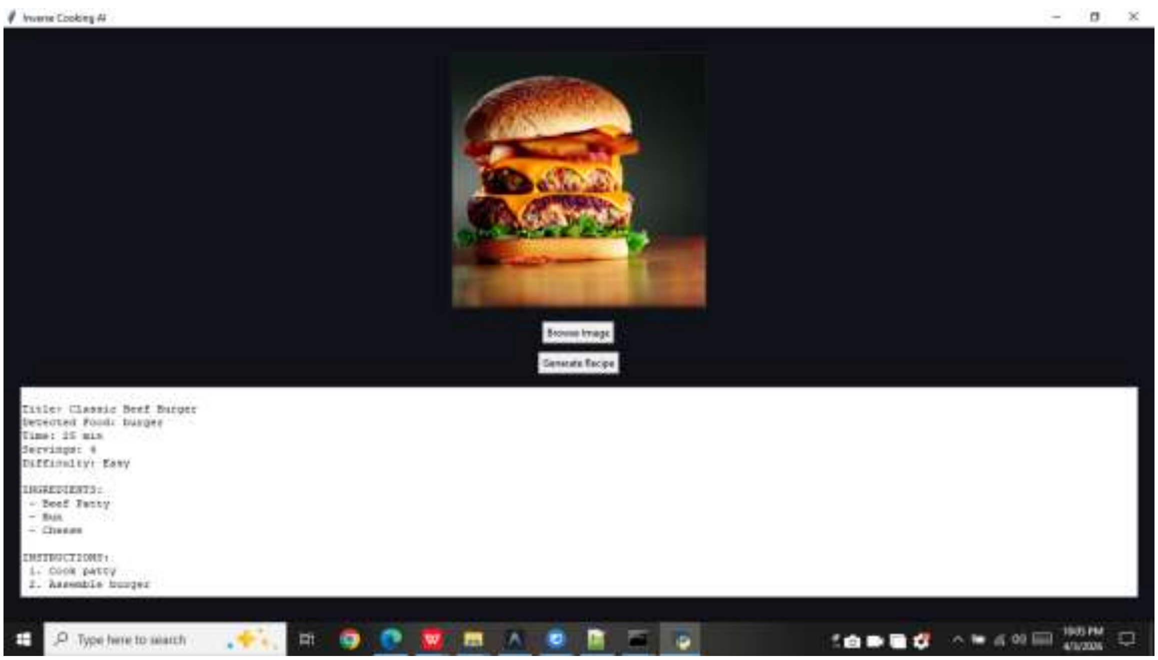
7. Advantages of Design

- Modular and scalable
- Easy to integrate AI models
- Lightweight and offline-capable
- User-friendly interface

SYSTEM DESIGN IMAGES



International Journal of Engineering Research and Science & Technology



VI. CONCLUSION

The Inverse Cooking AI system demonstrates a practical application of artificial intelligence concepts in the domain of food recognition and recipe generation. By integrating image processing techniques with a graphical user interface, the system provides an intuitive platform for users to generate recipes from food images. One of the key strengths of the system is its simplicity. Unlike complex deep learning models that require extensive datasets and computational resources, the proposed solution uses a lightweight approach that is easy to implement and operate. This makes it suitable for educational purposes and small-scale applications. The system successfully handles various image formats and ensures robustness through preprocessing techniques. The use of a structured food database enables quick retrieval of recipe information, enhancing user experience. Additionally, the GUI design ensures accessibility for users with minimal technical knowledge.

However, the system has certain limitations. The current prediction mechanism is based on filename matching, which may not accurately identify food items in real-world scenarios. Advanced systems use deep learning models to extract visual features and generate recipes, achieving higher accuracy and realism. Future enhancements may include integrating convolutional neural networks for image recognition, transformer-based models for recipe generation, and real-time image processing capabilities. The system can also be extended to support personalized recommendations based on dietary preferences. In conclusion, the project serves as a foundational model for inverse cooking applications. It highlights the potential of combining AI concepts with user-friendly interfaces to create innovative solutions. With further development, such systems can significantly improve the way users interact with food and cooking technologies.

REFERENCES

1. Salvador, A. et al. “Inverse Cooking: Recipe Generation from Food Images.” CVPR, 2019.
2. Salvador, A. et al. “Inverse Cooking System using Recipe1M Dataset.” arXiv, 2018.
3. Wang, H. et al. “Decomposing Generation Networks for Recipe Generation.” 2022.
4. Marin, J. et al. “Recipe1M+: Large Scale Dataset for Food Recognition.” 2019.
5. Sonika, R. et al. “Recipe Generator from Image using CNN and Transformers.” 2025.
6. Aman, A. et al. “Recipe Generation using CNN (ResNet-50).” 2024.
7. Sadique, P. M. et al. “Transformer-based Cooking Video Summarization.” 2025.
8. Salvador, A. et al. “Cross-modal Recipe Retrieval using Transformers.” 2021.
9. Chen, X. et al. “Food Image Recognition using Deep Learning.” 2023.
10. Liu, C. et al. “Multimodal Recipe Generation Systems.” 2022.
11. Zhang, Y. et al. “Attention-based Food Recognition Models.” 2023.
12. He, K. et al. “Deep Residual Learning for Image Recognition.” 2016.
13. Vaswani, A. et al. “Attention is All You Need.” 2017.
14. Tan, M. et al. “EfficientNet: Rethinking CNN Scaling.” 2019.
15. Goodfellow, I. et al. “Deep Learning (Book).” 2016.