Artificial Intelligence & Machine Learning project

Documentation

Introduction :

Project Title:

GrainPalette - A Deep Learning Odyssey In Rice Type Classification Through Transfer Learning

Members:

- Arikeri Usha Rani–Data Collection and Cleaning
- A Sai Hari Chandana–Model Design and Training
- A Chandra Vignesh–Evaluation and Optimization
- Adiki Manohar Kumare–Deployment and Integration

Project Overview

Purpose:

The purpose of this project is to develop an intelligent web-based tool that accurately classifies different rice grain varieties using image processing and deep learning techniques. This system aims to assist consumers, traders, and agricultural experts in identifying rice types quickly and reliably, reducing the dependency on manual inspection.

G oals:

- To develop a user-friendly rice classification system using machine learning.
- To reduce manual errors in rice variety identification.
- To create a scalable model that can be integrated into agricultural and quality control platforms.

Key Features:

- **Image-Based Classification**: Uses deep learning (CNN) for rice image analysis.
- **Multi-Class Support**: Classifies different rice types such as Basmati, Jasmine, Arborio, etc.
- **Visualization**: Shows prediction results, confidence scores, and graphs.
- **About Section**: Educates users on different rice types.
- **Web Interface**: Easy-to-use UI for uploading and viewing results.

Architecture

Frontend:

Technologies:

- **HTML/CSS** – Structure and styling
- **JavaScript** – Interaction and dynamic updates
- **Bootstrap/Tailwind CSS** – For responsive and clean UI
- **Jinja2 Templates**

Features:

- Image upload section
- Prediction results: Rice type + confidence score
- Charts/Graphs: Confidence distribution
- About page: Description of all rice types (Basmati, Arborio, Jasmine, etc.)
- Buttons: “Predict Again”, “Home”, “About”

Backend:

Technologies:

- **Python + Flask** – Web framework
- **TensorFlow/Keras or PyTorch** – Trained CNN model for classification
- **OpenCV / PIL** – Image processing
- **NumPy / Pandas** – Data manipulation

- **Matplotlib/Seaborn** – Charts

Features:

- Handle uploaded images
- Load and use CNN model to classify rice
- Generate visualizations (e.g., bar chart of prediction confidence)
- Return prediction result + chart to frontend

Model Integration:

A trained CNN model is saved as a `.h5` file and loaded in the Flask backend. When a user uploads an image, it is preprocessed and passed to the model. The model returns the predicted rice variety along with confidence scores. The result is then displayed on the frontend with visual feedback

Database:

Type:

- Rice_Image_Dataset

Stored Data:

- Uploaded image filenames or paths
- Prediction results (e.g., rice type, confidence)
- Timestamp of prediction
- User interaction logs

Use Cases for the Database:

- **User History Tracking**: Store and show previous predictions
- **Analytics**: Track popular rice types or model performance
- **Dataset Expansion**: Save new user images for retraining the model
- **Testing & Debugging**: Store misclassified samples for improvement.

Setup Instructions:

Prerequisites:

- Python3.
- Libraries:TensorFlow/Keras,NumPy,Pandas,OpenCV,Matplotlib,Streamlit
- GPU or Google Colab

STEP 1: Install Python and Required Packages

Ensure Python 3.x is installed.

Install dependencies using pip:

- pip install flask joblib numpy

STEP 2: Folder Structure:

- GrainPalette/
- │
- ├── app.py
- ├── model/
- ├── static/
- │ └── style.css
- ├── templates/
- │ ├── index.html
- │ ├── result.html
- │ └── about.html
- ├── utils/
- ├── uploads/
- └── requirements.txt

STEP 3: Running the App

Navigate to your project directory and run:

- python app.py
- Access the web app at: http://127.0.0.1:5000

How to Use:

- Open the browser
- Upload a rice grain image and click "Classify".
- View the predicted rice type, confidence score, and chart result.

Authentication :

- Authentication can be added using **Flask-Login** to manage user sessions and secure access.
- Users can **sign up, log in, and log out**, with credentials stored in a database

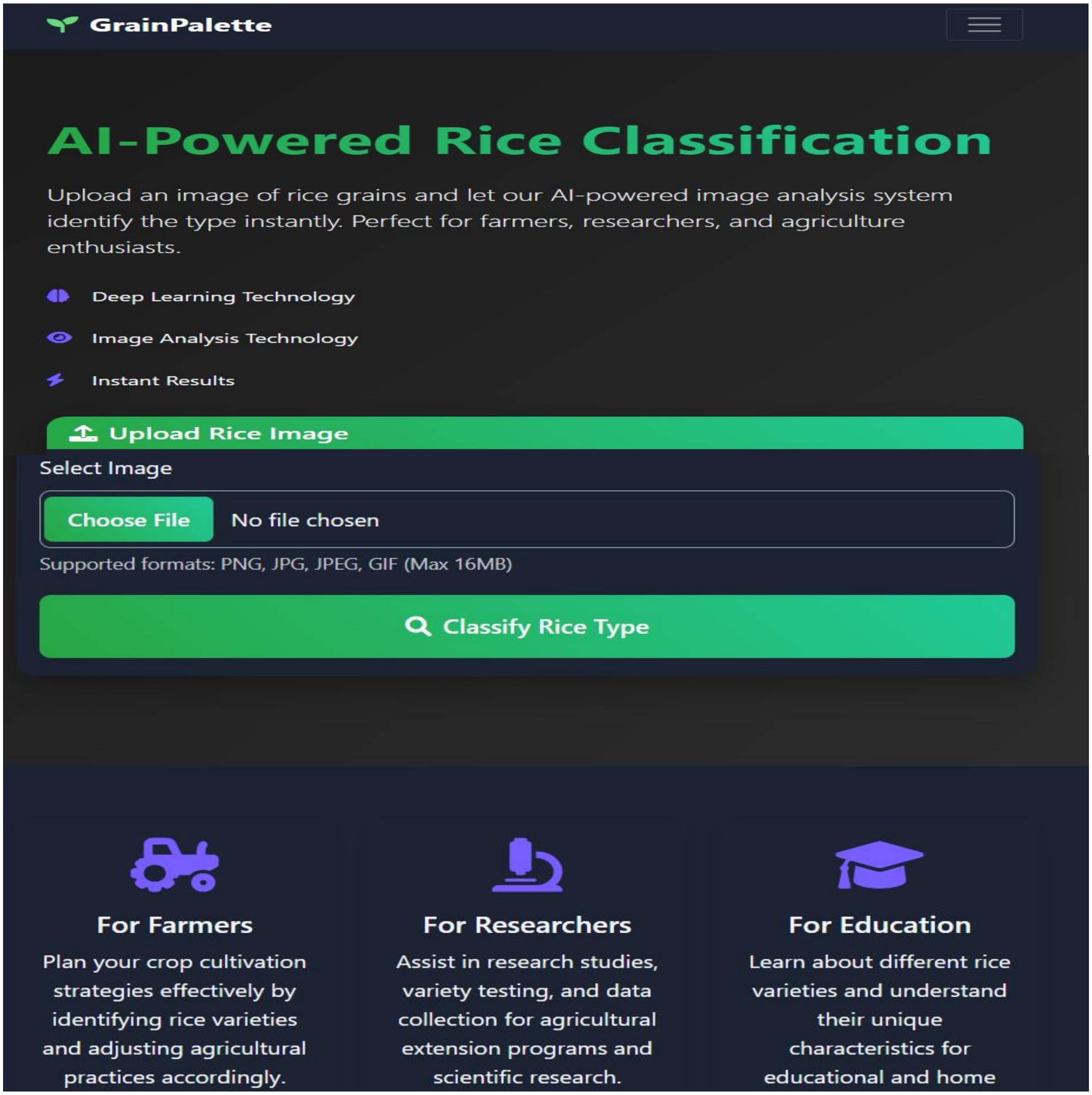
User Interface:

- Simple and clean form using HTML and Jinja2.
- Displays prediction result clearly.
- Can be extended with charts or model explanation

Testing:

- The system is tested by uploading different rice grain images and verifying the predicted labels and confidence scores.
- Functionality, model accuracy, UI responsiveness, and error handling are also validated through manual and edge-case testing.

Screenshots or Demo:



GrainPalette

AI-Powered Rice Classification

Upload an image of rice grains and let our AI-powered image analysis system identify the type instantly. Perfect for farmers, researchers, and agriculture enthusiasts.

- Deep Learning Technology
- Image Analysis Technology
- Instant Results

Upload Rice Image

Select Image

Choose File No file chosen

Supported formats: PNG, JPG, JPEG, GIF (Max 16MB)

Classify Rice Type

For Farmers

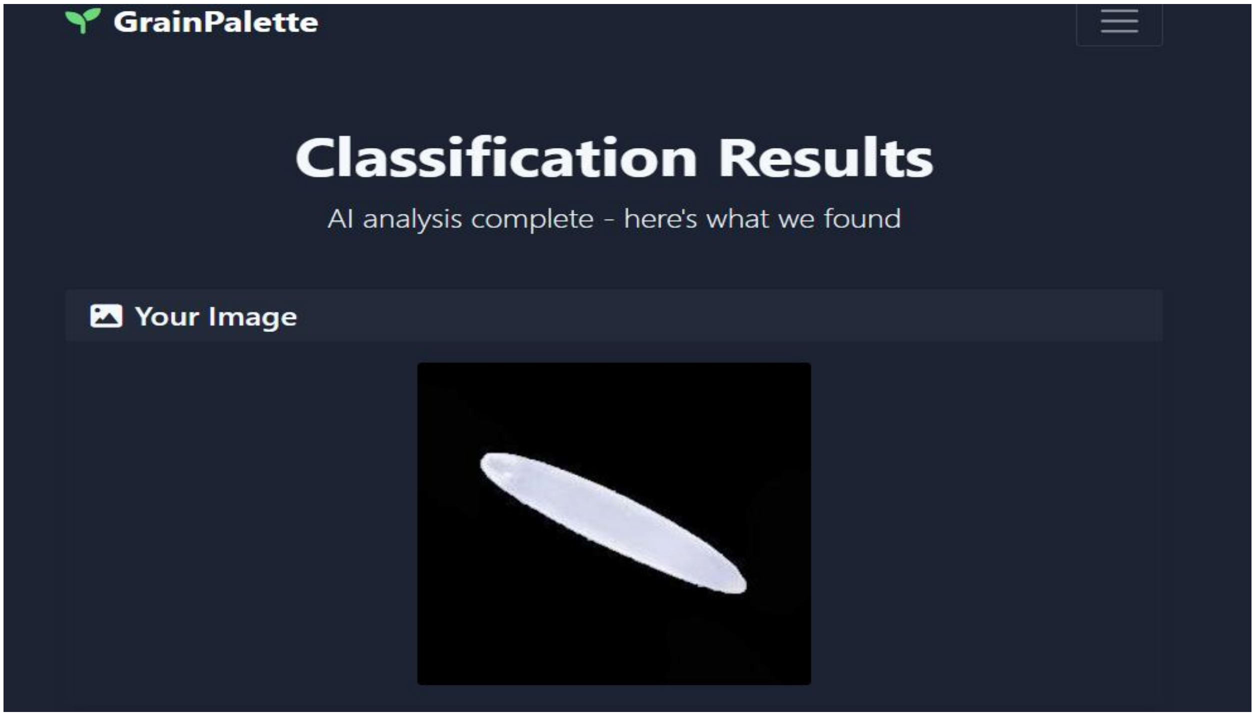
Plan your crop cultivation strategies effectively by identifying rice varieties and adjusting agricultural practices accordingly.

For Researchers

Assist in research studies, variety testing, and data collection for agricultural extension programs and scientific research.

For Education

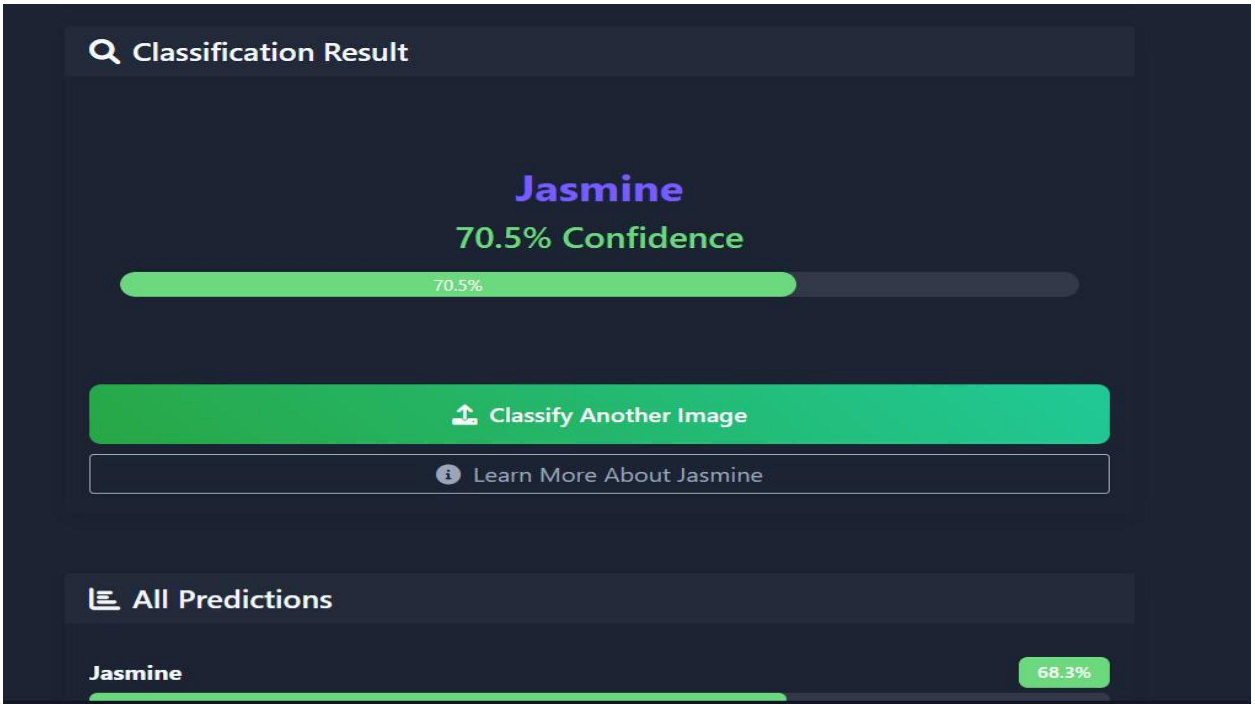
Learn about different rice varieties and understand their unique characteristics for educational and home



Classification Results

AI analysis complete - here's what we found

Your Image



Classification Result

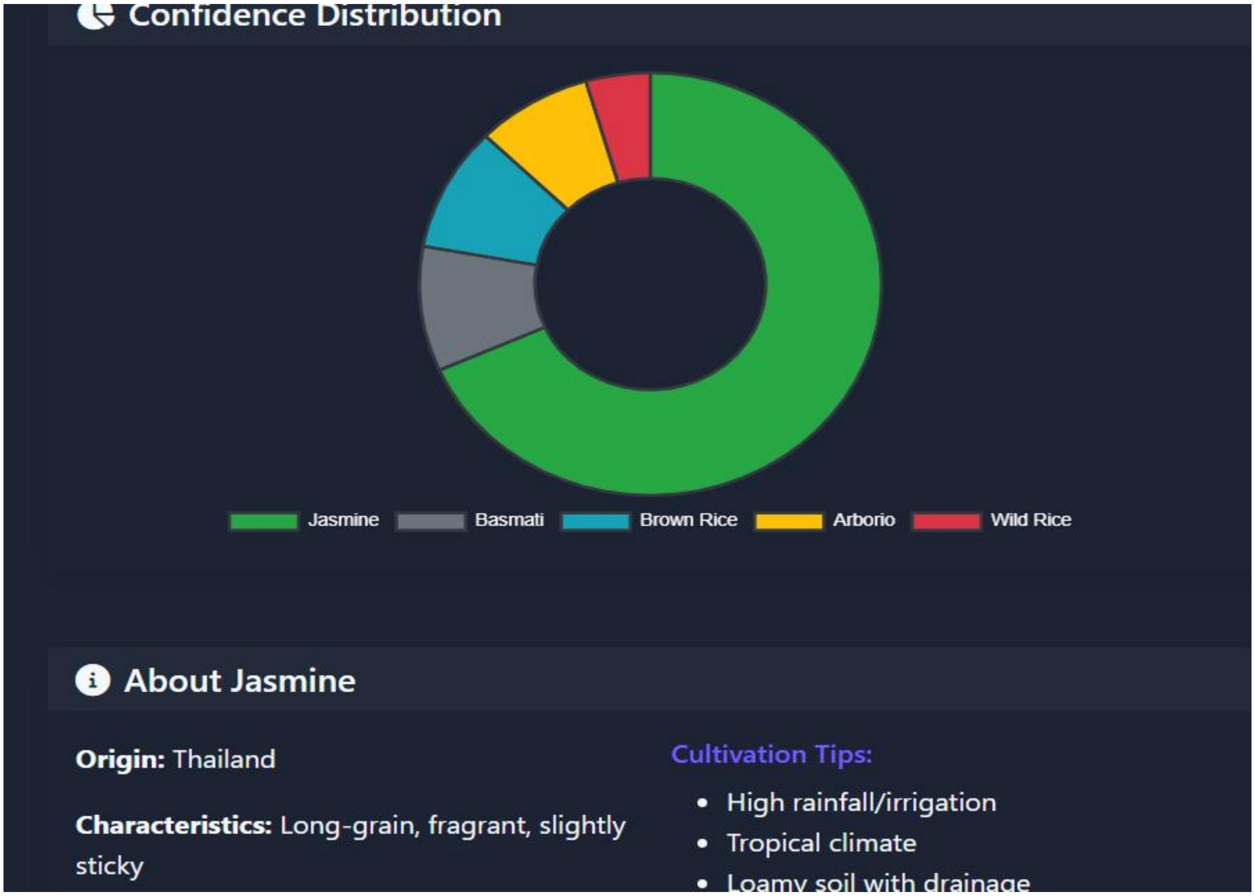
Jasmine

70.5% Confidence

70.5%

All Predictions

Jasmine 68.3%



About Jasmine

Origin: Thailand

Characteristics: Long-grain, fragrant, slightly sticky

Cultivation Tips:

- High rainfall/irrigation
- Tropical climate
- Loamy soil with drainage

Known Issues:

- **Misclassification** may occur with low-quality or unclear images.
- The model's accuracy may drop with **unseen rice varieties** not in the training dataset.

Future Enhancements:

- Add **user authentication** for personalized access and history tracking.
- Integrate a **larger dataset** to support more rice varieties and improve accuracy.
- Enable **real-time camera input** for live grain classification.