GrainPalette: Rice Type Classification Using Deep Learning

📌Project Overview

GrainPalette is a deep learning-based image classification project designed to identify five types of rice grains: **Arborio**, **Basmati**, **Ipsala**, **Jasmine**, and **Karacadag**. The model processes grain images and predicts their category with high accuracy. It is deployed via a Streamlit web interface, allowing users to upload an image and receive instant predictions.

📁Project Structure

```
GrainPalette/
├── app/                          # Streamlit app
│   └── predict_rice_type.py     # Streamlit web interface
│   └── bg.jpg                   # Optional background image
├── data/                        # Dataset folder
│   └── Rice_Image_Dataset/     # Original images grouped by class
│   └── Rice_Image_Dataset_split/ # Train/validation split
├── model/                       # Trained model storage
│   └── rice_classifier_model.h5
├── notebooks/                   # Jupyter notebooks
│   ├── 1_dataset_exploration.ipynb
│   ├── 2_data_preparation.ipynb
│   ├── 3_model_augmentation.ipynb
│   ├── 4_model_training.ipynb
│   ├── 5_model_evaluation.ipynb
│   ├── 6_model_saving_loading.ipynb
│   └── 7_model_inference.ipynb
├── saved_model/                # Streamlit deployment model location
│   └── rice_type_model.h5
├── .gitignore
├── README.md
└── requirements.txt
```

Model Architecture

- Input shape: (64, 64, 3)
- Layers:

- Conv2D(32) + MaxPooling
- Conv2D(64) + MaxPooling
- Flatten + Dense(128) + Dropout(0.5)
- Output: Dense(5, softmax)

Compiled with:

```
optimizer='adam'
loss='categorical_crossentropy'
metrics=['accuracy']
```

🧪 Dataset

- **Source**: Local rice grain image dataset
- **Classes**: Arborio, Basmati, Ipsala, Jasmine, Karacadag
- **Images**: Split into training and validation sets
- Training: \~60,000 images
- Validation: \~15,000 images

Data preprocessing:

```
ImageDataGenerator(rescale=1./255)
```

Augmentation used in training:

- Rotation, width/height shift, zoom, horizontal flip

📊 Training & Evaluation

- **Epochs**: 2–3 (adjusted for speed)
- **Input Size**: 64x64
- **Accuracy**: \~98–99% on validation set
- Model saved as: `rice_classifier_model.h5`

📖 Streamlit App (`predict_rice_type.py`)

- Upload an image
- Preprocesses it (resize to 64×64)
- Predicts class
- Displays confidence level

- Lists all class probabilities

Example usage:

```
streamlit run app/predict_rice_type.py
```

🧦Deployment

- Hosted on **Streamlit Cloud**
- Files required:
- `app/predict_rice_type.py`
- `saved_model/rice_type_model.h5`
- `requirements.txt`
- `.gitignore`, `README.md`

📌Requirements

- Python >= 3.8
- TensorFlow >= 2.8
- streamlit
- numpy, matplotlib, Pillow, etc.

`requirements.txt` example:

```
tensorflow
streamlit
numpy
matplotlib
Pillow
```

🔗Completed Notebooks

Notebook	Purpose
1_dataset_exploration.ipynb	Check classes and image counts
2_data_preparation.ipynb	Setup ImageDataGenerators
3_model_augmentation.ipynb	Augment training data
4_model_training.ipynb	Train CNN model

Notebook	Purpose
5_model_evaluation.ipynb	Evaluate metrics
6_model_saving_loading.ipynb	Save/load model
7_model_inference.ipynb	Test single image predictions

📊Inference Example

```
from tensorflow.keras.models import load_model
from tensorflow.keras.preprocessing import image

model = load_model("rice_classifier.h5")
img = image.load_img("path.jpg", target_size=(64, 64))
img_array = np.expand_dims(image.img_to_array(img)/255.0, axis=0)
pred = model.predict(img_array)
print("Predicted:", class_labels[np.argmax(pred)])
```

🕔Conclusion

GrainPalette successfully classifies rice grain images with a simple CNN. It features a clean UI and is ready for deployment or extension to other grain types.

Contact

- **Developer**: Bitta Saiumesh
- **Email**: saiumeshbitta@gmail.com
- **GitHub**: github.com/BittaSaiumesh/GrainPalette