

Data Collection and Preprocessing Phase

Date	08 July 2024
Team ID	SWTID1720201335
Project Title	Rice Type Classification Using CNN
Maximum Marks	6 Marks

Preprocessing Template

The images will be preprocessed by resizing, normalizing, augmenting, denoising, adjusting contrast, detecting edges, converting color space, cropping, batch normalizing, and whitening data. These steps will enhance data quality, promote model generalization, and improve convergence during neural network training, ensuring robust and efficient performance across various computer vision tasks.

Section	Description
Data Overview	<p>The dataset, sourced from Kaggle, Comprises images categorized into five classes of rice:</p> <ol style="list-style-type: none"> 1. arborio 2. basmati 3. ipsala 4. jasmine 5. karacadag.
Resizing	Images are resized to target size = (224,224).
Normalization	Images are normalized by dividing their pixel values by 255.0 to scale the data between 0 and 1.
Data Augmentation	<p>Rescaling: Pixel values are normalized by dividing by 255.0 to scale them between 0 and 1.</p> <p>Shear Range: Shearing transformations are applied with a range of 0.2.</p> <p>Width Shift Range: Horizontal shifts are applied with a range of 0.2.</p> <p>Height Shift Range: Vertical shifts are applied with a range of</p>

	<p>0.2.</p> <p>Rotation Range: Images are rotated within a range of 20 degrees.</p> <p>Zoom Range: Images are zoomed in and out with a range of 0.2.</p> <p>Horizontal Flip: Images are randomly flipped horizontally.</p>
Denoising	Noise is reduced using Non-Local Means Denoising with a filter strength of 10 for both color components and a template window size of 7
Data Preprocessing Code Screenshots	
Loading Data	<pre>[] direct="/content/Rice_Image_Dataset" direct=pathlib.Path(direct) direct PosixPath('/content/Rice_Image_Dataset')</pre> <pre>[] arborio = list(direct.glob('Arborio/*'))[:500] basmati = list(direct.glob('Basmati/*'))[:500] ipsala = list(direct.glob('Ipsala/*'))[:500] jasmine = list(direct.glob('Jasmine/*'))[:500] karacadag = list(direct.glob('Karacadag/*'))[:500] [] rice_names = { 'arborio' : arborio, 'basmati' : basmati, 'ipsala' : ipsala, 'jasmine' : jasmine, 'karacadag': karacadag } [] rice_index = { 'arborio' : 0, 'basmati' : 1, 'ipsala' : 2, 'jasmine' : 3, 'karacadag': 4 }</pre>
Resizing	<pre>[] X, y = [], [] for index, images in rice_names.items(): for image in images: img = cv2.imread(str(image)) resized_img = cv2.resize(img,(224,224)) X.append(resized_img) y.append(rice_index[index])</pre>
Normalization	<pre>X = np.array(X) X = X/255 y = np.array(y)</pre>

Data Augmentation	<pre> from keras.preprocessing.image import ImageDataGenerator datagen = ImageDataGenerator(rotation_range=20, width_shift_range=0.2, height_shift_range=0.2, shear_range=0.2, zoom_range=0.2, horizontal_flip=True, fill_mode='nearest') datagen.fit(X_train) history = model.fit(datagen.flow(X_train, y_train, batch_size=32), epochs=10, validation_data=(X_val, y_val)) </pre>
Denoising	<pre> denoised_img = cv2.fastNlMeansDenoisingColored(img, None, 10, 10, 7, 21) resized_img = cv2.resize(denoised_img, (224,224)) X.append(resized_img) </pre>
Edge Detection	-
Color Space Conversion	-
Image Cropping	-
Batch Normalization	-