Data Collection and Preprocessing Phase

Date	11 July 2024
Team ID	SWTID1720011518
Project Title	WCE Curated Colon Disease Classification using Deep Learning
Maximum Marks	6 Marks

Preprocessing

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	The images have been captured via Wireless Capsule Endoscopy (WCE). There are 3200 training images, 800 testing images and 2000 validation images. All are divided into four categories: Normal, Ulcerative Colitis, Polyps, Esophagitis.
Resizing	Resize images to a specified target size.
Normalization	Normalize pixel values to a specific range.
Data Augmentation	Apply augmentation techniques such as flipping, rotation, shifting, zooming, or shearing.
Denoising	Apply denoising filters to reduce noise in the images.
Edge Detection	Apply edge detection algorithms to highlight prominent edges in the images.

Color Space Conversion	Convert images from one color space to another.
Image Cropping	Crop images to focus on the regions containing objects of interest.
Batch Normalization	Apply batch normalization to the input of each layer in the neural network.

Data Preprocessing Code Screenshots

Loading Data	`# Define paths`
	`train_path = "/content/train"`
	`test_path = "/content/test"`
	`valid_path = "/content/val"`

Resizing	`# For testing and validation, only rescaling is applied`
	`test_datagen = ImageDataGenerator(rescale=1./255)`
	`valid_datagen = ImageDataGenerator(rescale=1./255)`

Normalization	`train_datagen = ImageDataGenerator(`
	`  rescale=1./255,`
	`  zoom_range=0.2,`
	`  shear_range=0.2,`
	`  preprocessing_function=preprocess_input  # VGG16 preprocessing (mean subtraction)`
	`)`

Data Augmentation	`train_datagen = ImageDataGenerator(`
	`  rescale=1./255,`
	`  zoom_range=0.2,`
	`  shear_range=0.2,`
	`  preprocessing_function=preprocess_input  # VGG16 preprocessing (mean subtraction)`
	`)`

Denoising	`def preprocess_image(img):`
	`  # Denoising with Gaussian blur`
	`  img = cv2.GaussianBlur(img, (5, 5), 0)`

Edge Detection	`# Edge detection with Canny edge detector` `edges = cv2.Canny(gray, 100, 200)`
Color Space Conversion	`# Convert to grayscale` `gray = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)`

Image Cropping	`# Image cropping (adjust cropping dimensions as needed)` `crop_img = img[50:150, 50:150]`

Batch Normalization	`# Flow data from directories`
	`train_generator = train_datagen.flow_from_directory(`
	`    train_path,`
	`    target_size=(224, 224),`
	`    batch_size=20,`
	`    class_mode='categorical'`