Data Collection and Preprocessing Phase

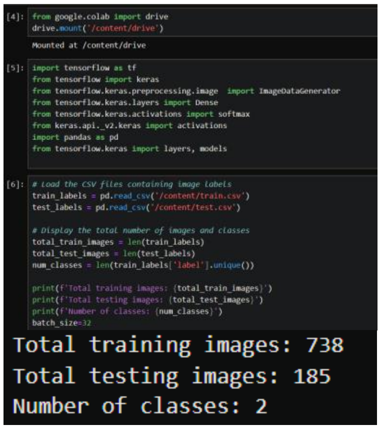
Date	12 july 2024
Team ID	SWTID1720084639
Project Title	**Beneath The Waves: Unraveling Coral Mysteries Through Deep Learning**
Maximum Marks	6 Marks

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	Our project leverages a dataset obtained from Kaggle, named '**Corals Image Classification'**. This dataset comprises 923 images featuring corals at various stages: Bleached and Healthy.
Resizing	ImageDataGenerator resizes images to a standard size (e.g., 299 x 299 pixels) during training for consistency and potentially improved model performance. This helps the model learn features effectively regardless of slight variations in original image sizes.
Normalization	ImageDataGenerator (rescale=l./255) already normalizes the pixel values to the specific range of 0 to I
Data Augmentation	ImageDataGenerator to artificially create variations of coral images during training. This includes techniques like random rotations and horizontal flipping. This helps the model learn robust features and improve its generalizability to unseen coral images.

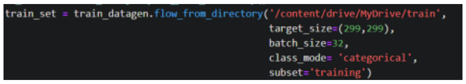
<table>
<tr><td>Denoising</td><td>NA</td></tr>
<tr><td>Edge Detection</td><td>NA</td></tr>
<tr><td>Color Space Conversion</td><td>NA</td></tr>
<tr><td>Image Cropping</td><td>NA</td></tr>
<tr><td>Batch Normalization</td><td>-</td></tr>
<tr><td colspan="2"><b>Data Preprocessing Code Screenshots</b></td></tr>
<tr><td>Loading Data</td><td>



```
[4]: from google.colab import drive
     drive.mount('/content/drive')

     Mounted at /content/drive

[5]: import tensorflow as tf
     from tensorflow import keras
     from tensorflow.keras.preprocessing.image  import ImageDataGenerator
     from tensorflow.keras.layers import Dense
     from tensorflow.keras.activations import softmax
     from keras.api._v2.keras import activations
     import pandas as pd
     from tensorflow.keras import layers, models

[6]: # Load the CSV files containing image labels
     train_labels = pd.read_csv('/content/train.csv')
     test_labels = pd.read_csv('/content/test.csv')

     # Display the total number of images and classes
     total_train_images = len(train_labels)
     total_test_images = len(test_labels)
     num_classes = len(train_labels['label'].unique())

     print(f'Total training images: {total_train_images}')
     print(f'Total testing images: {total_test_images}')
     print(f'Number of classes: {num_classes}')
     batch_size=32
```

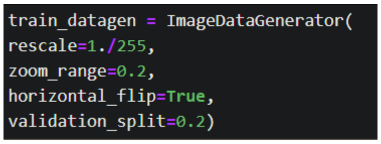
```
Total training images: 738
Total testing images: 185
Number of classes: 2
```

</td></tr>
<tr><td>Resizing</td><td>



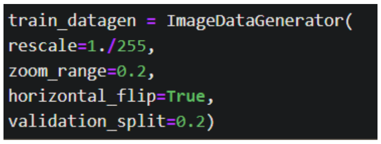
```
train_set = train_datagen.flow_from_directory('/content/drive/MyDrive/train',
                                              target_size=(299,299),
                                              batch_size=32,
                                              class_mode= 'categorical',
                                              subset='training')
```

</td></tr>
<tr><td>Normalization</td><td>



```
train_datagen = ImageDataGenerator(
rescale=1./255,
zoom_range=0.2,
horizontal_flip=True,
validation_split=0.2)
```

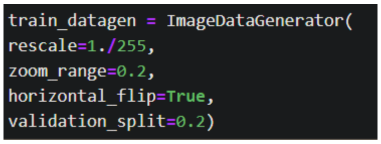
</td></tr>
<tr><td>Data Augmentation</td><td>



```
train_datagen = ImageDataGenerator(
rescale=1./255,
zoom_range=0.2,
horizontal_flip=True,
validation_split=0.2)
```

</td></tr>
</table>

Denoising	NA
Edge Detection	NA
Color Space Conversion	NA
Image Cropping	NA

Batch Normalization	
	train_datagen = ImageDataGenerator(
	rescale=1./255,
	zoom_range=0.2,
	horizontal_flip=True,
	validation_split=0.2)