Data Collection and Preprocessing Phase

Date	12 July 2024
Team ID	SWTID1720351492
Project Name	Covid Vision: Advanced COVID-19 Detection from Lung X-Rays with Deep Learning
Maximum Marks	3 Marks

Preprocessing

This project involves data analysis and preprocessing of a COVID-19 radiography dataset. It includes visualizing class distribution, resizing images to 224x224 pixels, normalizing pixel values between 0 and 255, label encoding the classes (COVID, normal), and reshaping the processed images and class labels into NumPy arrays for classification.

Section	Description
Data Overview	data analysis and preprocessing on a COVID-19 radiography dataset, including visualization of class distribution and image processing.
Resizing	Resize images to a specified target size (224,224,3)
Normalization	Normalize pixel values to a specific range (0, 255)
Label encoding	Label encoding the class values (covid, normal) for classification
Dimension Reshaping	Reshaping the x, y NumPy arrays which has the processed image and class, respectively.

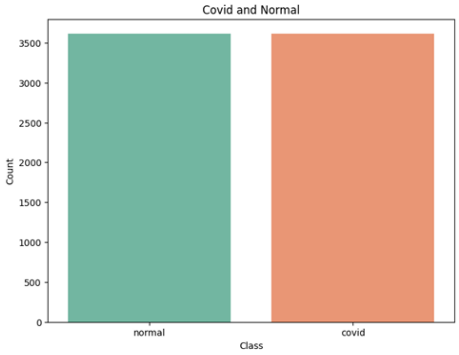
Data Preprocessing Code Screenshots

Loading Data

```
[10]: df.info()
```

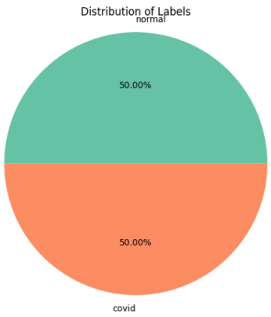
```
<class 'pandas.core.frame.DataFrame'>
Index: 7232 entries, 1265 to 6460
Data columns (total 2 columns):
 #   Column  Non-Null Count  Dtype
---  ------  --------------  -----
 0   image   7232 non-null   object
 1   class   7232 non-null   object
dtypes: object(2)
memory usage: 169.5+ KB
```

```
[11]: plt.figure(figsize=(8, 6))
sns.countplot(x='class', data=df, palette='Set2')
plt.title('Covid and Normal')
plt.xlabel('Class')
plt.ylabel('Count')
plt.show()
```




```
[12]: label_counts = df['class'].value_counts()
plt.figure(figsize=(6, 6))
plt.pie(label_counts, labels=label_counts.index, autopct='%1.2f%%', colors=sns.
  ↪color_palette('Set2'))
plt.title('Distribution of Labels')
plt.axis('equal')
plt.show()
```

‘



```
[13]: def convert_image(image):
          img = cv2.imread(image)
          img = img_to_array(img)
          img = cv2.resize(img, (224, 224))
          return (img)

[14]: df['processed_image'] = df['image'].apply(convert_image)

[15]: df.head()

[15]:                                                  image   class  \
      1265  /kaggle/input/covid19-radiography-database/COV…  normal
      1553  /kaggle/input/covid19-radiography-database/COV…  normal
      4437  /kaggle/input/covid19-radiography-database/COV…   covid
      3771  /kaggle/input/covid19-radiography-database/COV…   covid
      3249  /kaggle/input/covid19-radiography-database/COV…  normal
```



```
                                                processed_image
1265  [[[226.53745, 226.53745, 226.53745], [206.3273…
1553  [[[2.171875, 2.171875, 2.171875], [2.171875, 2…
4437  [[[46.793156, 46.793156, 46.793156], [16.27830…
3771  [[[0.0, 0.0, 0.0], [0.0, 0.0, 0.0], [0.0, 0.0,…
3249  [[[3.502232, 3.502232, 3.502232], [5.5855637, …
```

Resizing

```
[13]: def convert_image(image):
          img = cv2.imread(image)
          img = img_to_array(img)
          img = cv2.resize(img, (224, 224))
          return (img)

[14]: df['processed_image'] = df['image'].apply(convert_image)
```

Normalization

```
[22]: x_train = x_train / 255
      x_test_scaled = x_test / 255
      x_val = x_val / 255
```

Label encoding

```
[16]: le = LabelEncoder()
      df['processed_class'] = le.fit_transform(df['class'])
      df['processed_class'].unique()

[16]: array([1, 0])
```

Dimension Reshaping

```
[18]: x = np.stack(df['processed_image'].values)
      y = np.array(df['processed_class']).reshape(-1, 1)

[19]: print(len(x))
      print(x.shape)
      print(y.shape)

      7232
      (7232, 224, 224, 3)
      (7232, 1)
```