

pneumonia_detection

September 14, 2023

```
[ ]: # importing libraries
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
%matplotlib inline
import seaborn as sns
import tensorflow as tf
from sklearn.metrics import confusion_matrix, confusion_matrix, \
    classification_report, auc, roc_curve
from tensorflow.keras.preprocessing.image import ImageDataGenerator
from tensorflow.keras import models
from tensorflow.keras import optimizers
from tensorflow.keras import layers
from tensorboard.plugins.hparams import api as hp
import os

[ ]: # creating dataframes
test_df = ('archive_10/chest_xray/chest_xray/test')
train_df = ('archive_10/chest_xray/chest_xray/train')
val_df = ('archive_10/chest_xray/chest_xray/val')

[ ]: # Getting the counts for each dataframe
print('Test set:')
num_pneumonia_test = len(os.listdir(os.path.join(test_df, 'PNEUMONIA')))
num_normal_test = len(os.listdir(os.path.join(test_df, 'NORMAL')))
print(f"PNEUMONIA={num_pneumonia_test}")
print(f"NORMAL={num_normal_test}")

print("Train set:")
num_pneumonia_train = len(os.listdir(os.path.join(train_df, 'PNEUMONIA')))
num_normal_train = len(os.listdir(os.path.join(train_df, 'NORMAL')))
print(f"PNEUMONIA={num_pneumonia_train}")
print(f"NORMAL={num_normal_train}")
print("\n")

print("Val set:")
num_pneumonia_val = len(os.listdir(os.path.join(val_df, 'PNEUMONIA')))
```

```

num_normal_val = len(os.listdir(os.path.join(val_df, 'NORMAL')))
print(f"PNEUMONIA={num_pneumonia_val}")
print(f"NORMAL={num_normal_val}")
print("\n")

```

Test set:
PNEUMONIA=390
NORMAL=234
Train set:
PNEUMONIA=3876
NORMAL=1342

Val set:
PNEUMONIA=9
NORMAL=9

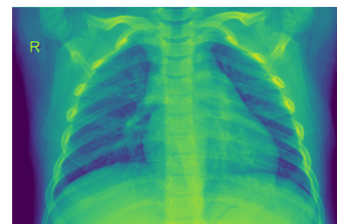
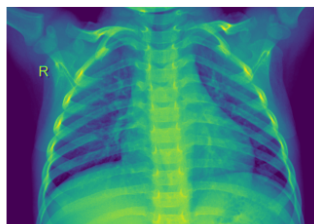
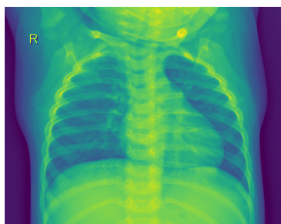
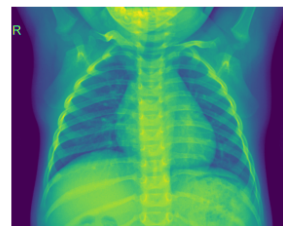
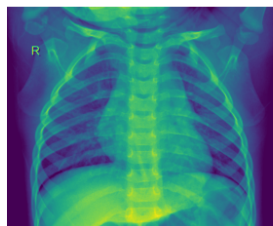
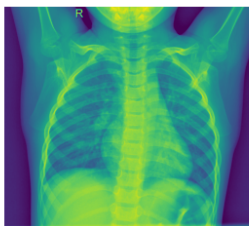
```

[ ]: #Illustrating examples of a normal xray
print('Examples of a normal xray:\n-----')
normal = os.listdir('archive_10/chest_xray/train/NORMAL')
normal_directory = 'archive_10/chest_xray/train/NORMAL'

plt.figure(figsize = (15,10))
for i in range(6):
    plt.subplot(3, 3, i+1)
    img = plt.imread(os.path.join(normal_directory, normal[i]))
    plt.imshow(img)
    plt.axis('off')

```

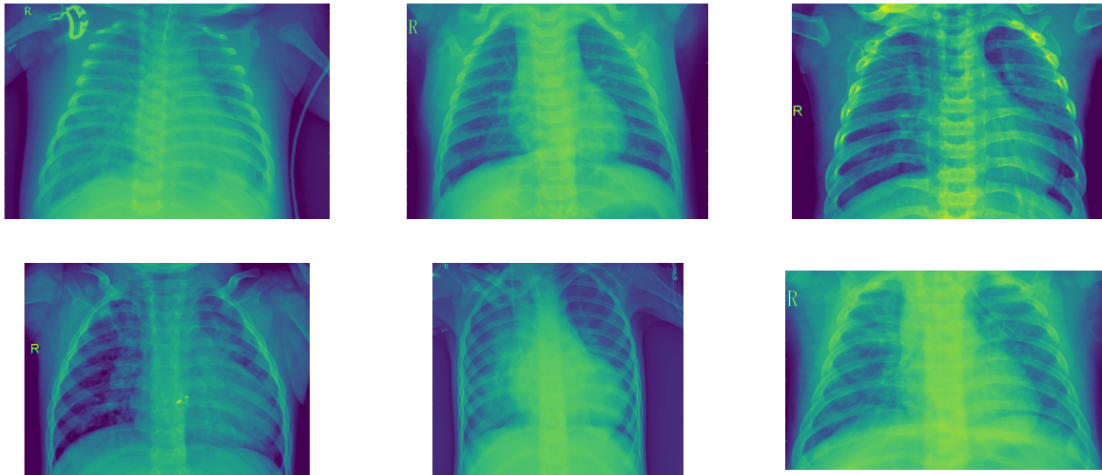
Examples of a normal xray:



```
[ ]: #illustrating examples of a pneumonia xray
print('Examples of a pneumonia xray:\n-----')
pneumonia = os.listdir('archive_10/chest_xray/train/PNEUMONIA')
pneumonia_directory = 'archive_10/chest_xray/train/PNEUMONIA'

plt.figure(figsize = (15,10))
for i in range(6):
    plt.subplot(3, 3, i+1)
    img = plt.imread(os.path.join(pneumonia_directory, pneumonia[i]))
    plt.imshow(img)
    plt.axis('off')
```

Examples of a pneumonia xray:



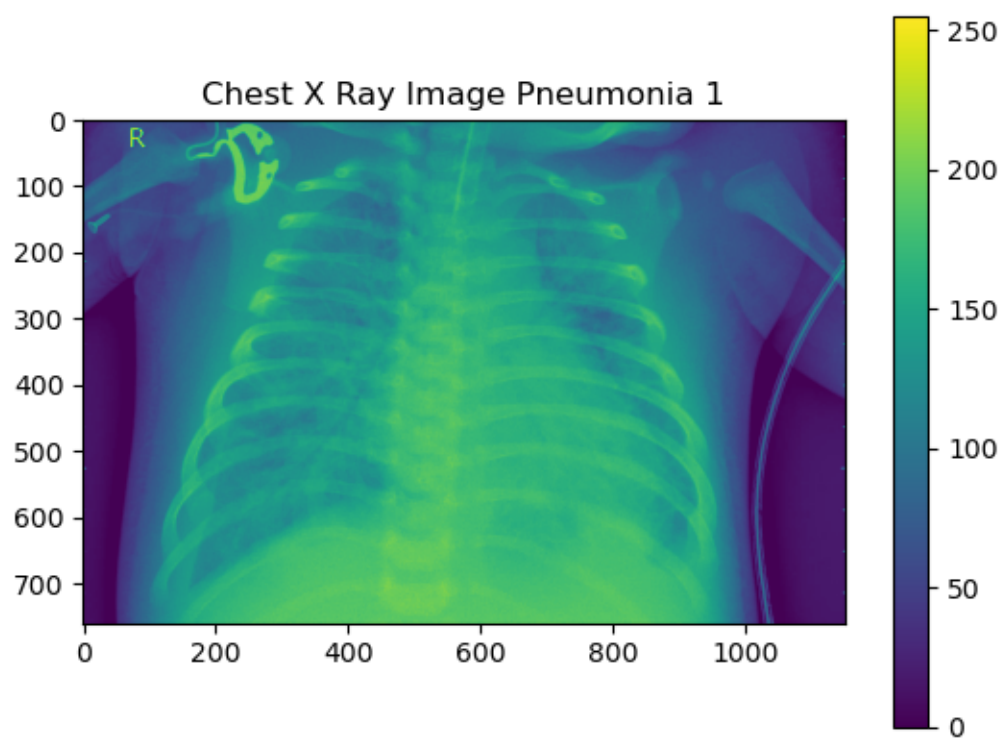
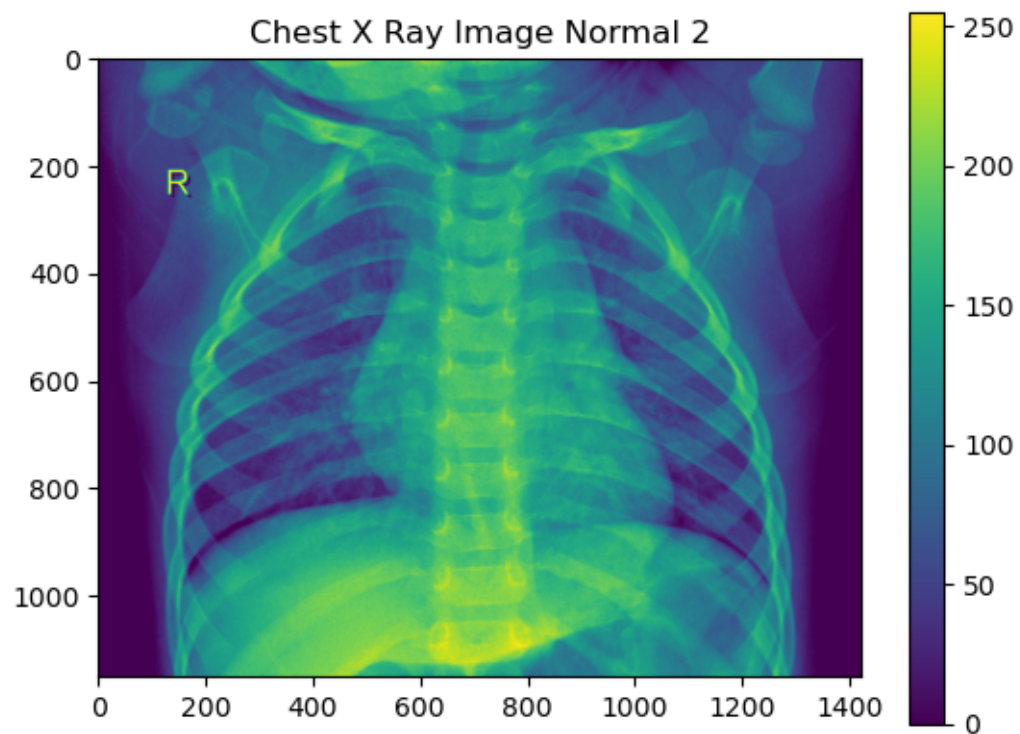
```
[ ]: norm_1 = os.listdir("archive_10/chest_xray/train/NORMAL")[0]
norm_2 = os.listdir("archive_10/chest_xray/train/NORMAL")[1]
pnu_1 = os.listdir("archive_10/chest_xray/train/PNEUMONIA")[0]
pnu_2 = os.listdir("archive_10/chest_xray/train/PNEUMONIA")[1]

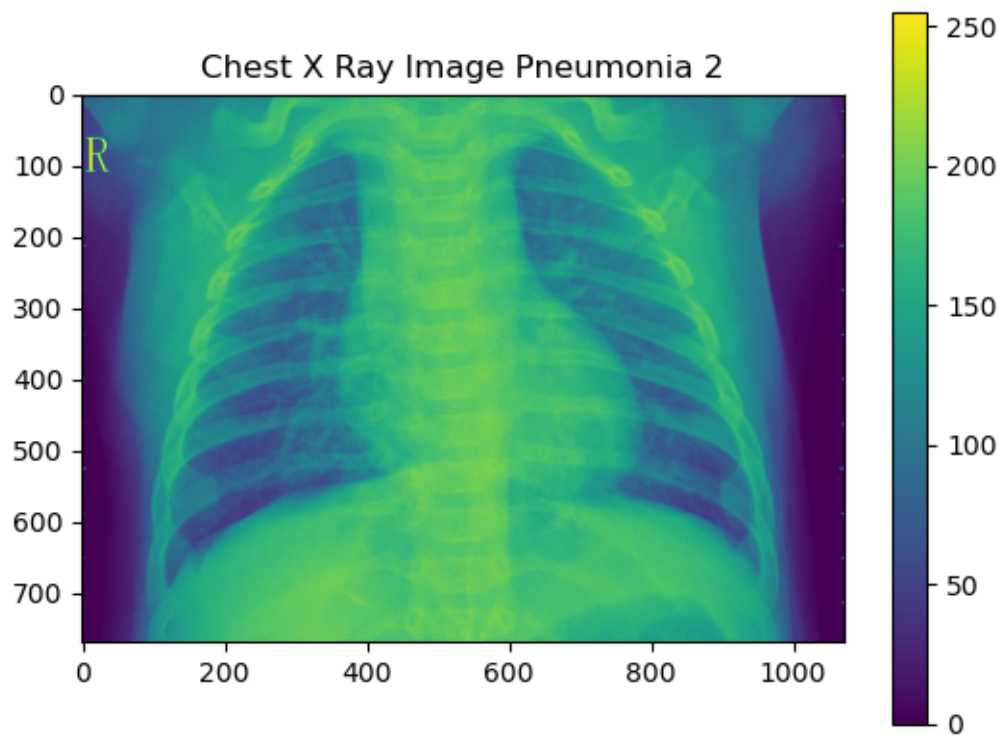
sample_norm_1 = plt.imread(os.path.join(normal_directory, norm_1))
sample_norm_2 = plt.imread(os.path.join(normal_directory, norm_2))
sample_pnu_1 = plt.imread(os.path.join(pneumonia_directory, pnu_1))
sample_pnu_2 = plt.imread(os.path.join(pneumonia_directory, pnu_2))

plt.imshow(sample_norm_1)
plt.colorbar()
```

```
plt.title('Chest X Ray Image Normal 1')
plt.show()
plt.imshow(sample_norm_2)
plt.colorbar()
plt.title('Chest X Ray Image Normal 2')
plt.show()
plt.imshow(sample_pnu_1)
plt.colorbar()
plt.title('Chest X Ray Image Pneumonia 1')
plt.show()
plt.imshow(sample_pnu_2)
plt.colorbar()
plt.title('Chest X Ray Image Pneumonia 2')
plt.show()
```



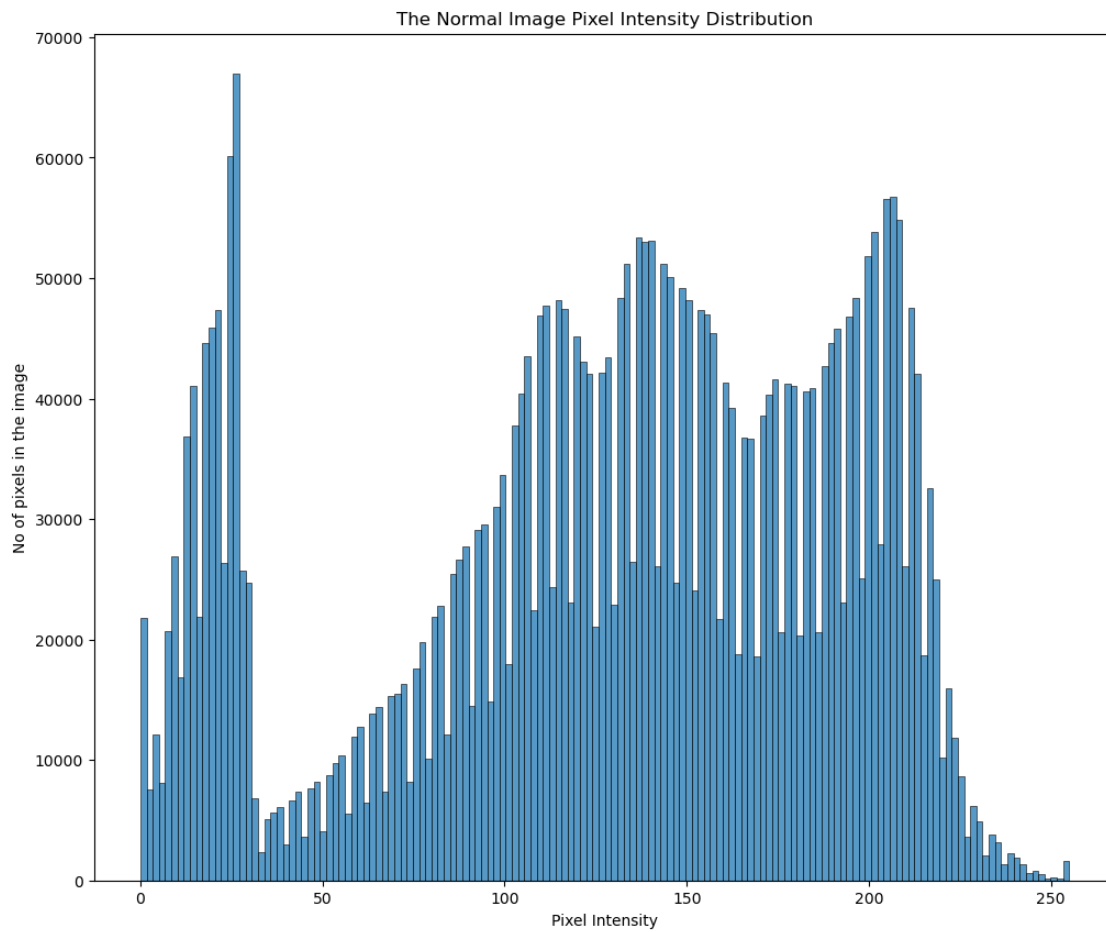


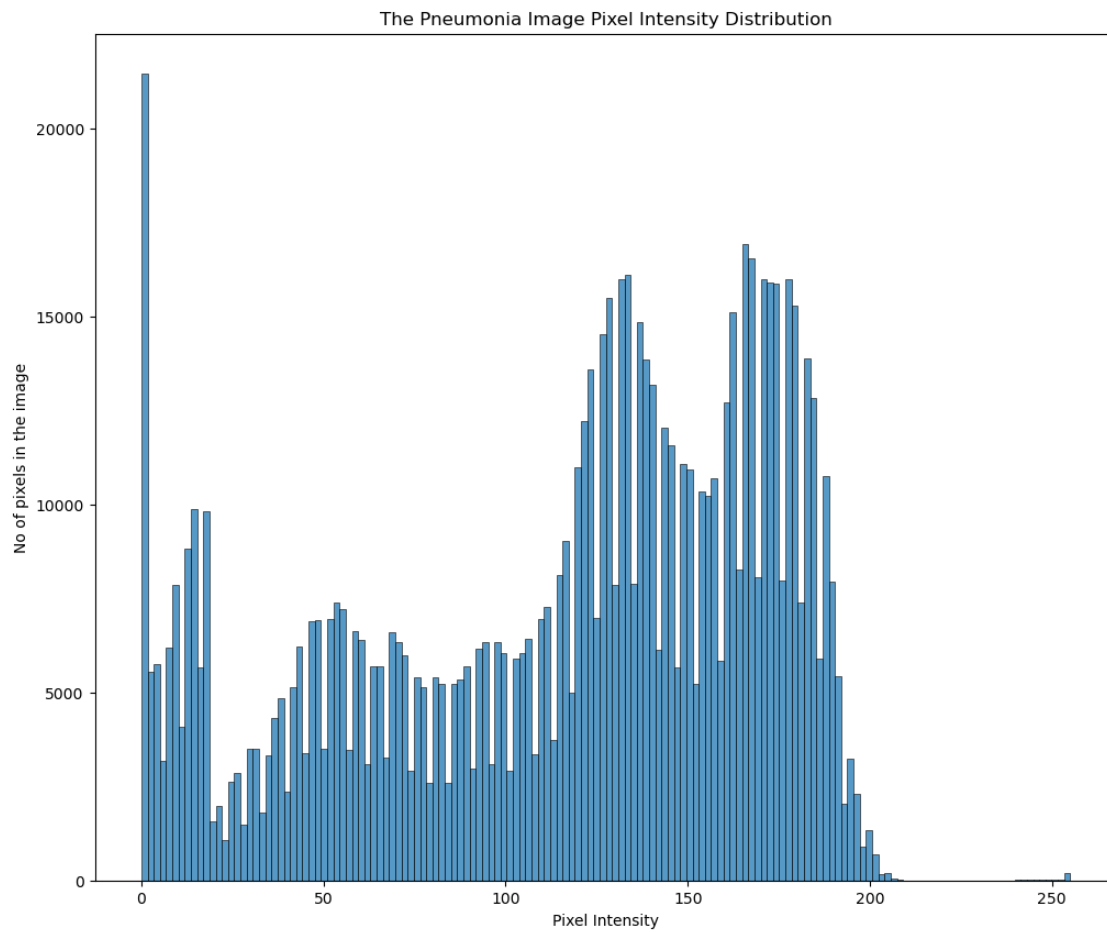


```
[ ]: #plotting pixel intensities
def plot_image_histogram(img_1, img_2):
    fig, ax = plt.subplots(figsize = (12, 10))
    sns.histplot(img_1.ravel(), bins = 150,)
    plt.title('The Normal Image Pixel Intensity Distribution')
    plt.xlabel('Pixel Intensity')
    plt.ylabel('No of pixels in the image')
    plt.show()

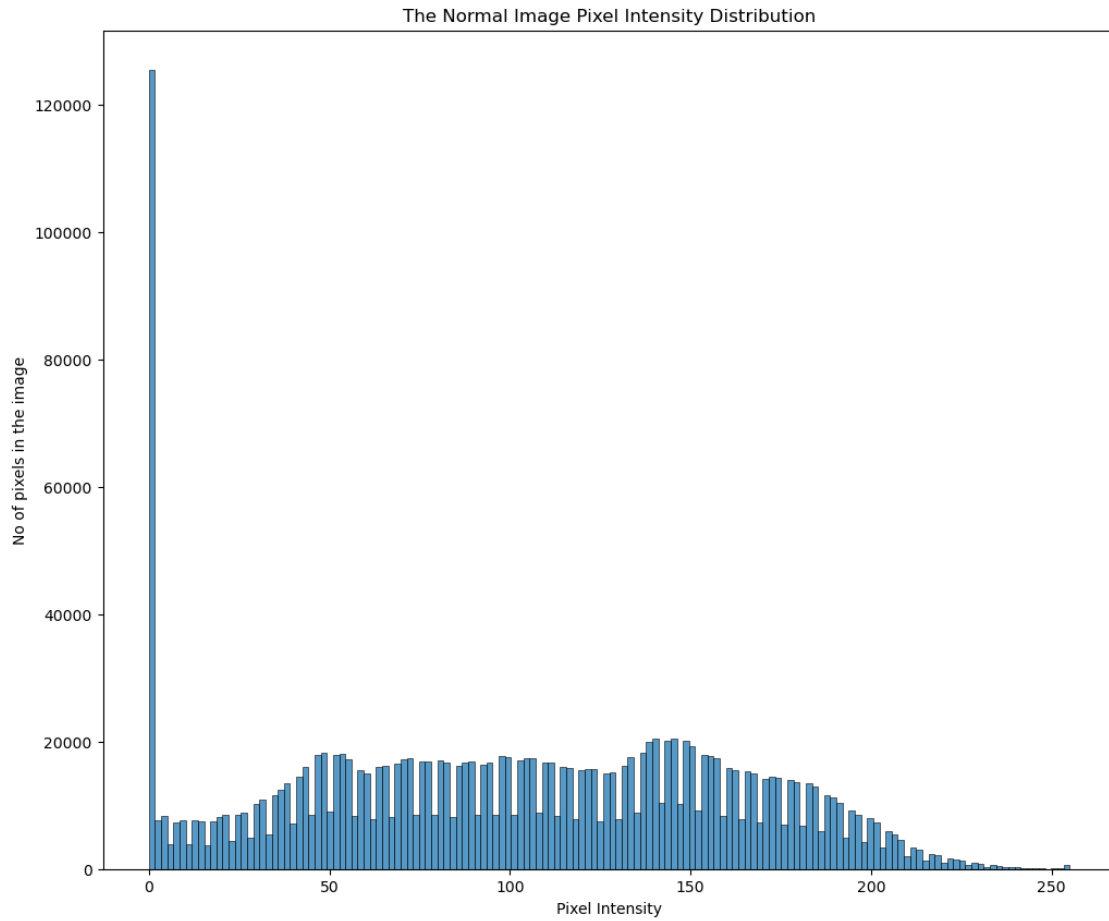
    fig, ax = plt.subplots(figsize = (12, 10))
    sns.histplot(img_2.ravel(), bins = 150)
    plt.title('The Pneumonia Image Pixel Intensity Distribution')
    plt.xlabel('Pixel Intensity')
    plt.ylabel('No of pixels in the image')
    plt.show()

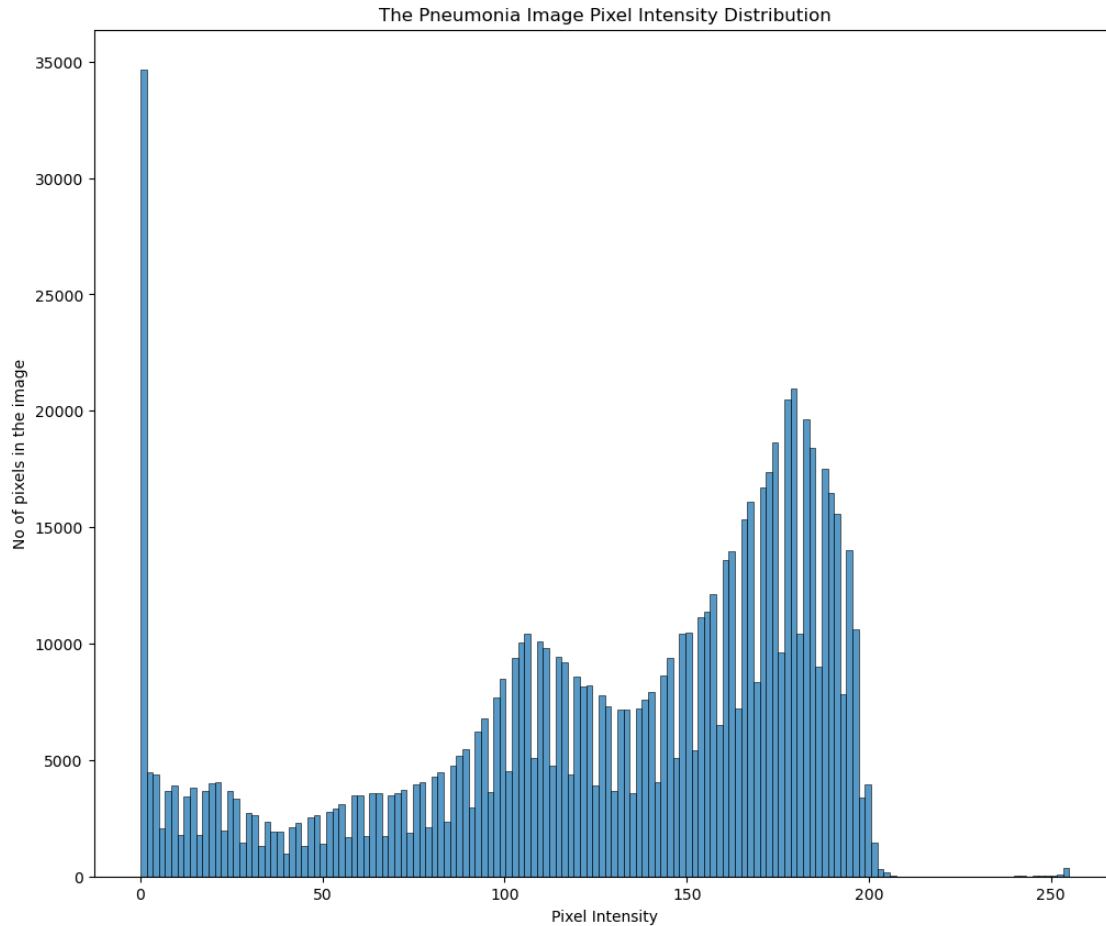
plot_image_histogram(sample_norm_1, sample_pnu_1)
```





```
[ ]: plot_image_histogram(sample_norm_2, sample_pnu_2)
```



```
[ ]: #data nomarilization
train_datagen = ImageDataGenerator(rescale = 1./255)
val_datagen = ImageDataGenerator(rescale = 1./255)

#Using a generator for the images to work with.
train_generator = train_datagen.flow_from_directory(train_df,
                                                    target_size = (224, 224),
                                                    batch_size=32,
                                                    class_mode = 'binary',
                                                    shuffle=True)

validation_generator = val_datagen.flow_from_directory(val_df,
                                                       target_size = (224, 224),
                                                       batch_size = 32,
                                                       class_mode = 'binary',
                                                       shuffle=True)
```

Found 5216 images belonging to 2 classes.

Found 16 images belonging to 2 classes.

```
[ ]: #information classification check
train_generator.class_indices
validation_generator.class_indices
```

```
[ ]: {'NORMAL': 0, 'PNEUMONIA': 1}
```

```
[ ]: num_pneumonia = []
num_normal = []
#Calculating class weights to deal with data imbalance
weight_for_0 = num_pneumonia / (num_normal + num_pneumonia)
weight_for_1 = num_normal / (num_normal + num_pneumonia)
class_weight = {0: weight_for_0, 1: weight_for_1}

print(f"Weight for class 0: {weight_for_0:.2f}")
print(f"Weight for class 1: {weight_for_1:.2f}")
```

TypeError

Traceback (most recent call last)

c:

↪ \Users\hp\Documents\Flatiron\phase_4_project\Pneumonia-Detection\pneumonia_detection.
↪ ipynb Cell 11 line 4

↪

↪

---->

↪

↪

TypeError: unsupported operand type(s) for /: 'list' and 'list'

```
[ ]:
```