Lung Detection Final Project

Danny Mathieson

Project Contents

3. Add a dense layer at the end of the pre-trained model, followed by a dropout layer and try various combinations to optimize accuracy
4. Create a final output using the softmax activation function
5. Change the loss function to categorical cross-entropy
6. Use Adam as the optimizer
7. Use Early Stopping on the validation loss with a patience of 2 epochs to prevent overfitting
8. Use 15 epochs with a batch size of 7 - tinker with these to optimize accuracy
9. Train using an image generator and test the accuracy on the test data at each epoch
10. Plot the training & validation accuracy & loss
11. Observe metrics Precision, Recall, F1-Score for all classes on both grayscale & color models - determine if the classes are good.

Final Step

1. Compare all of the models on the basis of accuracy, precision, recall, f1-score

Section 1: Project Setup

1. Import Necessary Libraries

In [2]:
```
import os
import shutil
import numpy as np
from tensorflow.keras.utils import load_img, img_to_array, array_to_img
from tensorflow.keras.preprocessing.image import ImageDataGenerator
import matplotlib.pyplot as plt

# Set Style for Matplotlib plots
plt.style.use('ggplot')
```

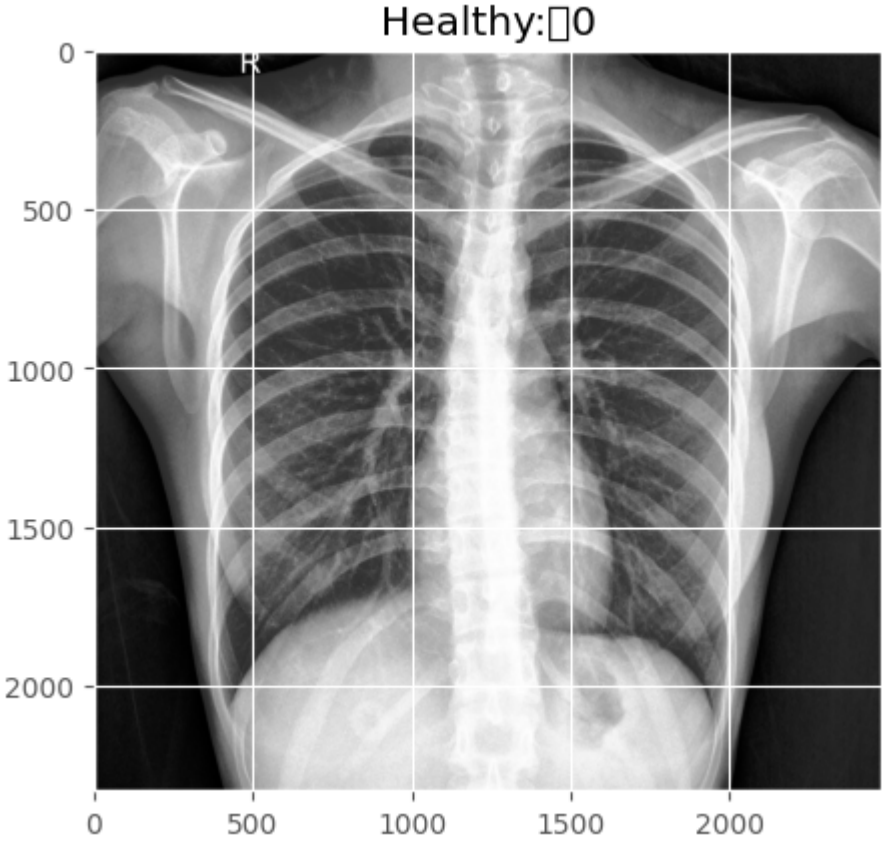
```
2023-03-30 18:41:53.695806: I tensorflow/core/platform/cpu_feature_guard.cc:193] This
TensorFlow binary is optimized with oneAPI Deep Neural Network Library (oneDNN) to us
e the following CPU instructions in performance-critical operations:  AVX2 FMA
To enable them in other operations, rebuild TensorFlow with the appropriate compiler
flags.
```

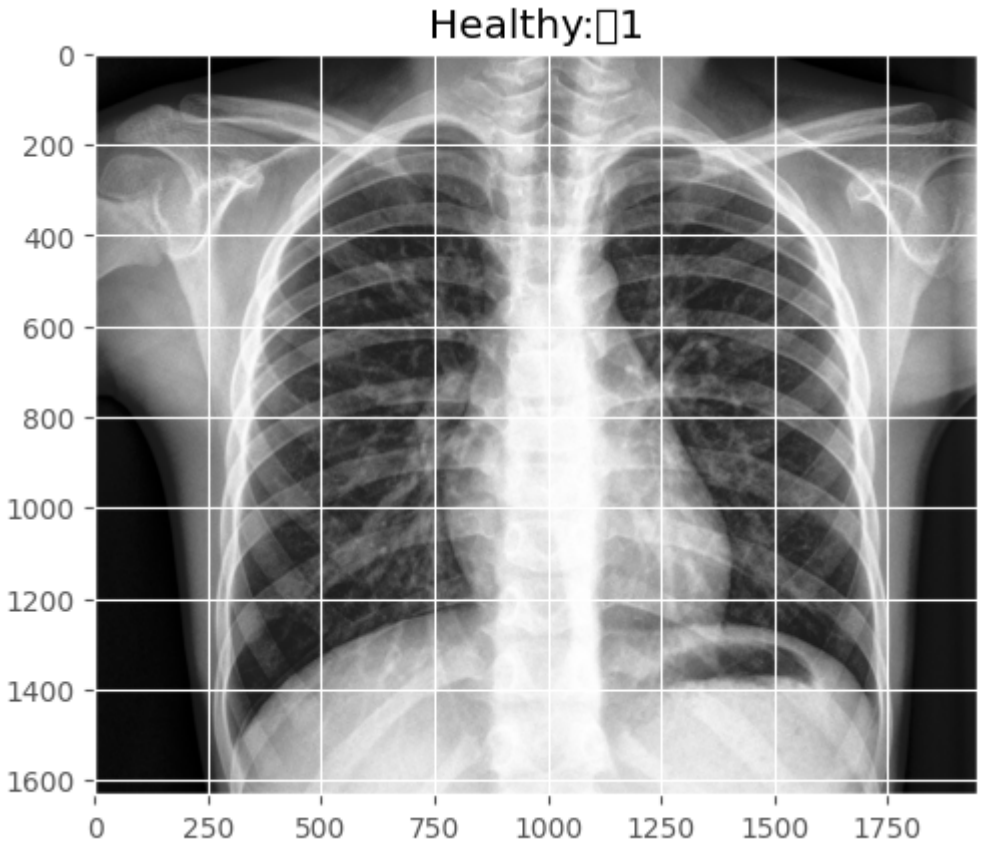
In [3]:
```
shutil.unpack_archive('./images/Dataset_Detection_of_Lung_Infection.zip', './images')
os.rename('./images/data/test/healthy/','./images/data/test/Healthy/')
```

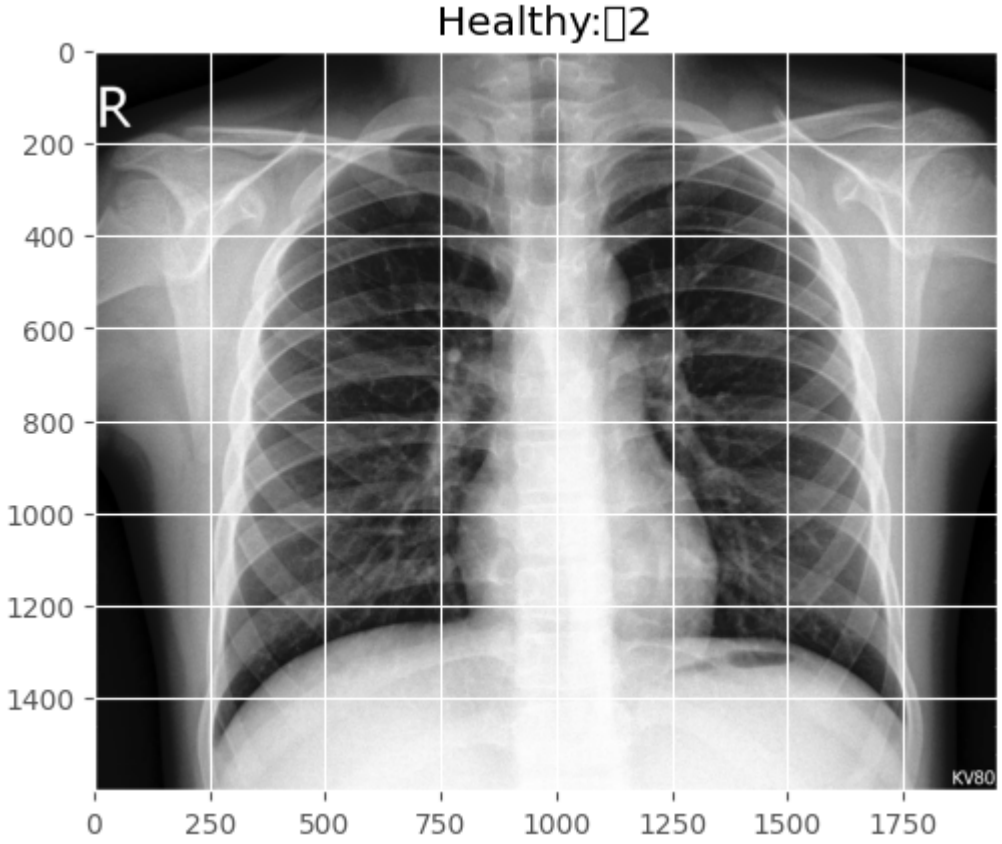
2. Plot sample images for all classes

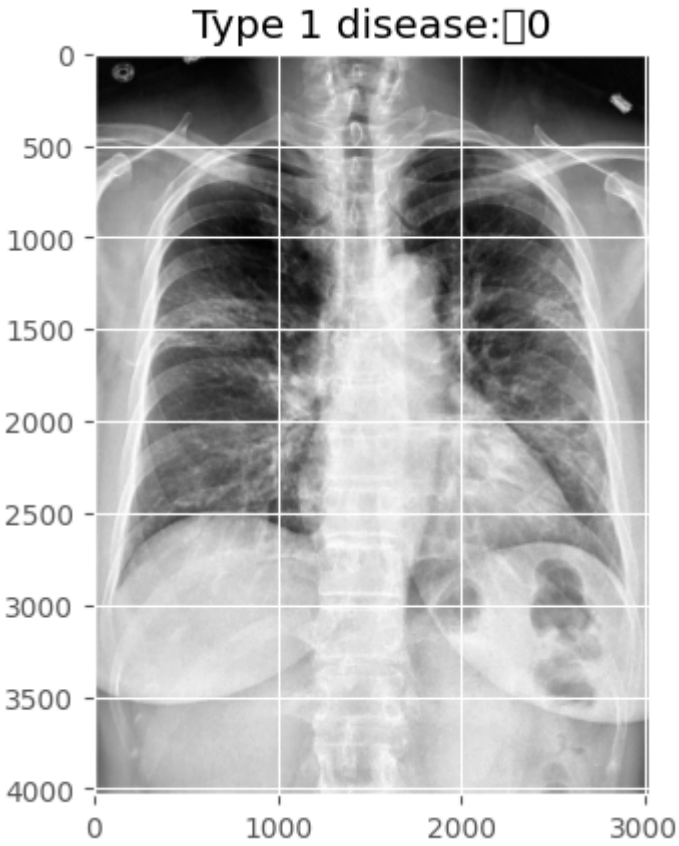
In [4]:
```
classes = ['Healthy', 'Type 1 disease', 'Type 2 disease']
train_path = './images/data/train/'
test_path = './images/data/test/'
sample_images = 3

for c in classes:
    for i in range(sample_images):
        img = load_img(train_path + c + '/' + os.listdir(train_path + c)[i])
        x = img_to_array(img)
        plt.title(f'{c}:\t{i}')
        plt.imshow(x/255.)
        plt.show()
```

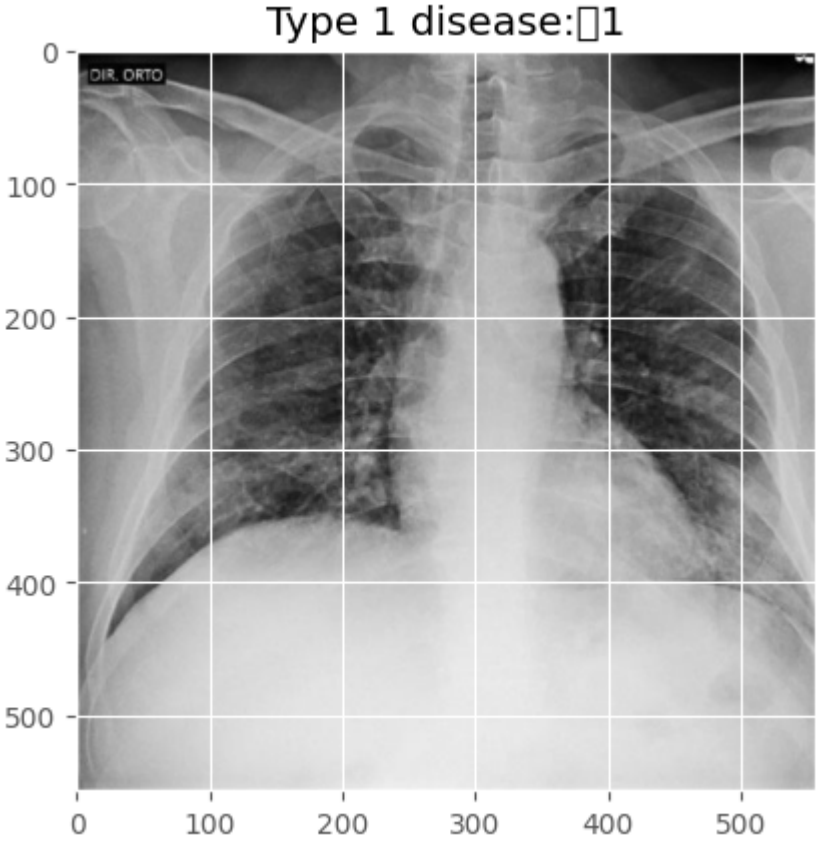
```
/usr/local/lib/python3.10/site-packages/IPython/core/pylabtools.py:151: UserWarning:
Glyph 9 (	) missing from current font.
  fig.canvas.print_figure(bytes_io, **kw)
```

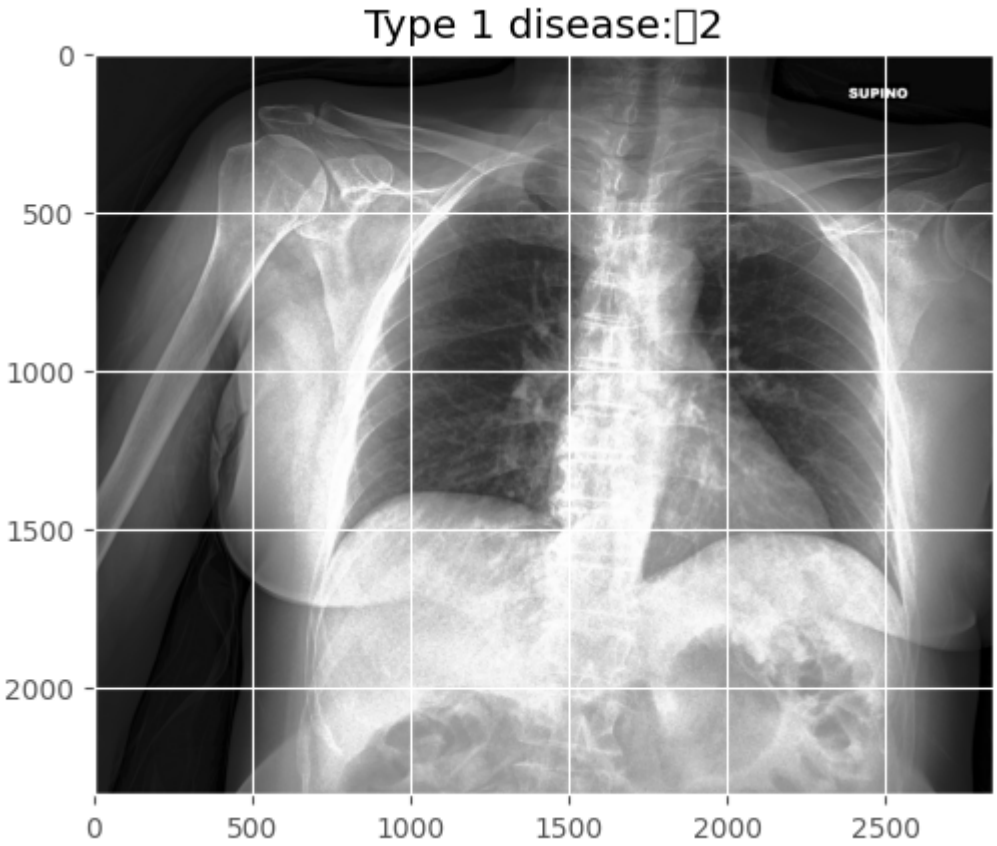

Healthy:□0
R
0
500
1000
1500
2000
0
500
1000
1500
2000

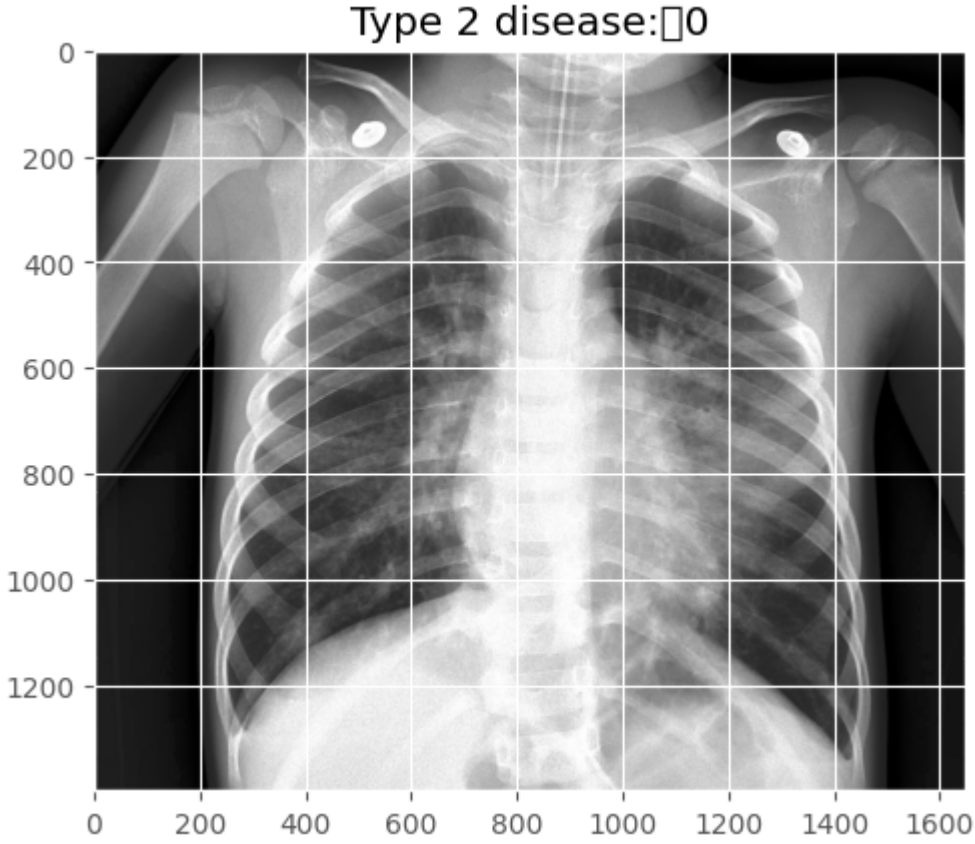

Healthy:□1
0
200
400
600
800
1000
1200
1400
1600
0
250
500
750
1000
1250
1500
1750

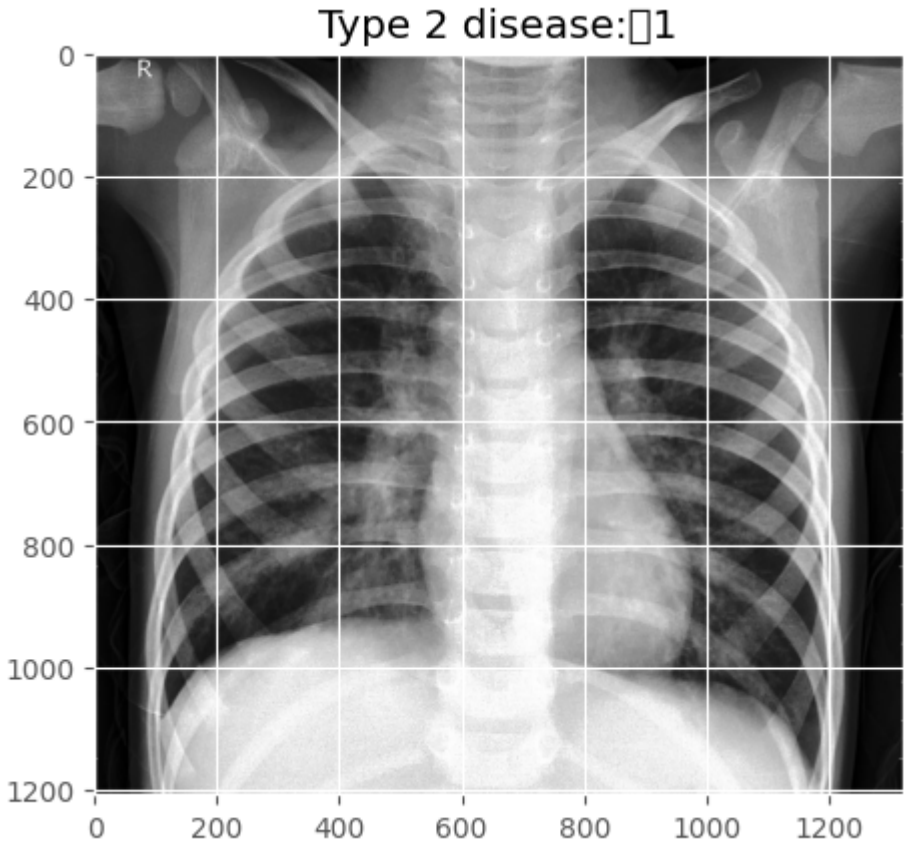

Healthy:□2
R
KV80
0
200
400
600
800
1000
1200
1400
0
250
500
750
1000
1250
1500
1750

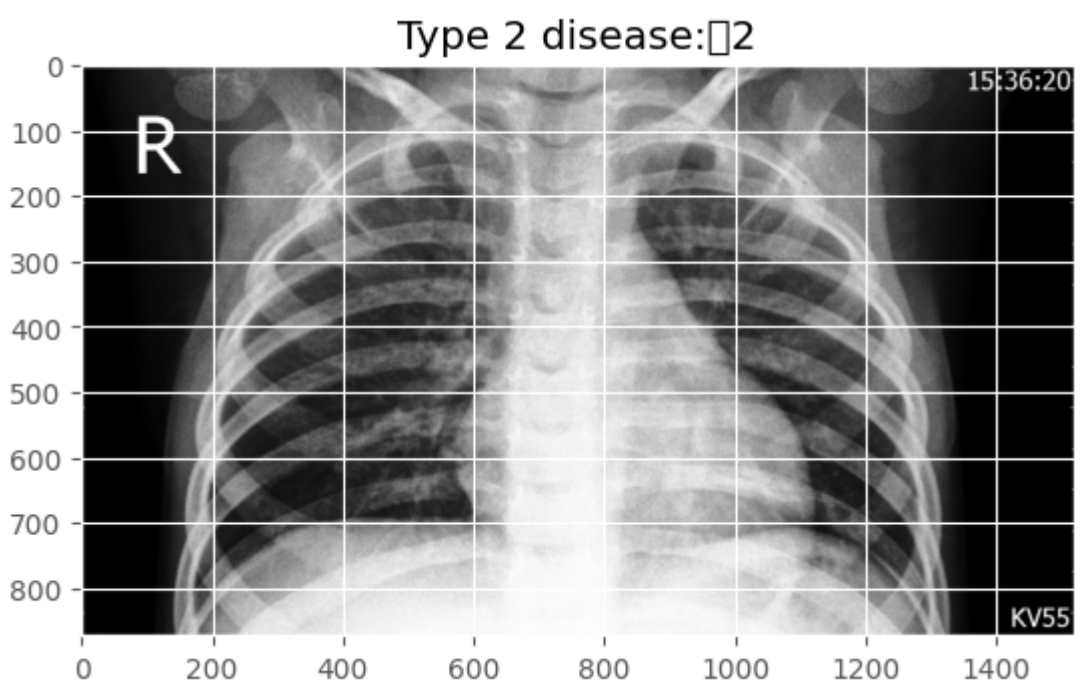

Type 1 disease:□0
0
500
1000
1500
2000
2500
3000
3500
4000
0
1000
2000
3000


Type 1 disease:□1
DIR. ORTO
0
100
200
300
400
500


Type 1 disease:□2
SUPINO
0
500
1000
1500
2000
2500


Type 2 disease:□0
0
200
400
600
800
1000
1200
0
200
400
600
800
1000
1200
1400
1600


Type 2 disease:□1
R
0
200
400
600
800
1000
1200
0
200
400
600
800
1000
1200



3. Plot the distribution of images across classes

In [5]:
```
# Retrieve the number of images for each class in the test & train set
def get_num_images(path):
    num_images = {}
    for c in classes:
        num_images[c] = len(os.listdir(path + c))
    return num_images

class_counts = {}
for path in [('Train',train_path), ('Test',test_path)]:
    class_counts[path[0]] = get_num_images(path[1])

print(class_counts)
```

```
{'Train': {'Healthy': 70, 'Type 1 disease': 111, 'Type 2 disease': 70}, 'Test': {'Hea
lthy': 20, 'Type 1 disease': 26, 'Type 2 disease': 20}}
```

In [6]:
```
# Plot the number of images for each class in the test & train set
for path in class_counts.keys():
    plt.bar(class_counts[path].keys(), class_counts[path].values())
    plt.title(f'Number of Images in {path} Set')
    plt.show()
```

4. Build Data Augmentation for the training data with translation, rescale, and flip - Rescale images to 48x48

In [7]:
```
# Create an image generator to augment the images in the training set
train_gen = ImageDataGenerator(
    rescale=1./255,
    rotation_range=40,
    width_shift_range=0.2,
    height_shift_range=0.2,
    shear_range=0.2,
    zoom_range=0.2,
    horizontal_flip=True,
    fill_mode='nearest'
```

```
)

# Point the training generator to the training set to create augmented images at trai
train_generator = train_gen.flow_from_directory(
    train_path,
    target_size=(48,48),
    batch_size=16,
    class_mode='categorical',
    shuffle=True
)
```

```
Found 251 images belonging to 3 classes.
```

5. Build Data Augmentation for the test data with translation, rescale, and flip - Rescale images to 48x48

In [8]:
```
# Create an image generator to augment the images in the test set
test_gen = ImageDataGenerator(
    rescale=1./255
)

# Point the test generator to the test set to create augmented images at test time
test_generator = test_gen.flow_from_directory(
    test_path,
    target_size=(48,48),
    batch_size=16,
    class_mode='categorical',
    shuffle=True
)
```

```
Found 66 images belonging to 3 classes.
```

6. Make a function to read directly from the train and test folders

In [9]:
```
# Function to read directly from the train & test generators
def read_from_generator(generator):
    X = []
    y = []
    for i in range(len(generator)):
        X.append(generator[i][0])
        y.append(generator[i][1])
    X = np.concatenate(X, axis=0)
    y = np.concatenate(y, axis=0)
    return X, y
```

Section 2: Build Initial CNN

1. Build a CNN with different filters, max pooling, dropout, and batch normalization layers

2. Use ReLU as an activation function

In [37]:
```
# import the necessary packages
from tensorflow.keras.models import Sequential, Model, load_model
from tensorflow.keras.layers import Conv2D, MaxPooling2D, Flatten, Dense, Dropout, Ba
from tensorflow.keras.optimizers import Adam, RMSprop
from tensorflow.keras.callbacks import EarlyStopping, ModelCheckpoint
from tensorflow.keras.activations import relu, softmax
from tensorflow.keras.losses import categorical_crossentropy
from tensorflow.keras.metrics import categorical_accuracy, Precision, Recall
from livelossplot import PlotLossesKerasTF
from sklearn.metrics import classification_report, precision_score, recall_score, f1_

# Define Tracking Metrics during Training
METRICS = [
    categorical_accuracy,
```

```
    Precision(name='precision'),
    Recall(name='recall')
]
```

In [11]:
```
# Build the model
model = Sequential()
model.add(Input(shape=(48,48,3)))

# Convolutional Layer 1
model.add(Conv2D(64, (3,3), activation=relu, padding='same'))
model.add(BatchNormalization())
model.add(MaxPooling2D((2,2)))
model.add(Dropout(0.2))

# Convolutional Layer 2
model.add(Conv2D(32, (3,3), activation=relu, padding='same'))
model.add(BatchNormalization())
model.add(MaxPooling2D((2,2)))
model.add(Dropout(0.2))

# Convolutional Layer 3
model.add(Conv2D(16, (3,3), activation=relu, padding='same'))
model.add(BatchNormalization())
model.add(MaxPooling2D((2,2)))
model.add(Dropout(0.2))

# Flatten the output of the convolutional layers
model.add(Flatten())

# Dense Layer 1
model.add(Dense(128, activation=relu))
model.add(Dropout(0.2))

# Dense Layer 2
model.add(Dense(64, activation=relu))
model.add(Dropout(0.2))

# Output Layer
model.add(Dense(3, activation=softmax))
```

3. Use Categorical Cross Entropy as a loss function

4. Use rmsprop as the optimizer

In [12]:
```
model.compile(
    loss=categorical_crossentropy,
    optimizer=RMSprop(learning_rate=0.001),
    metrics=METRICS
)
```

5. Use an early stopping with a patience of 2 epochs on validation loss or validation accuracy

6. Use 10 epochs

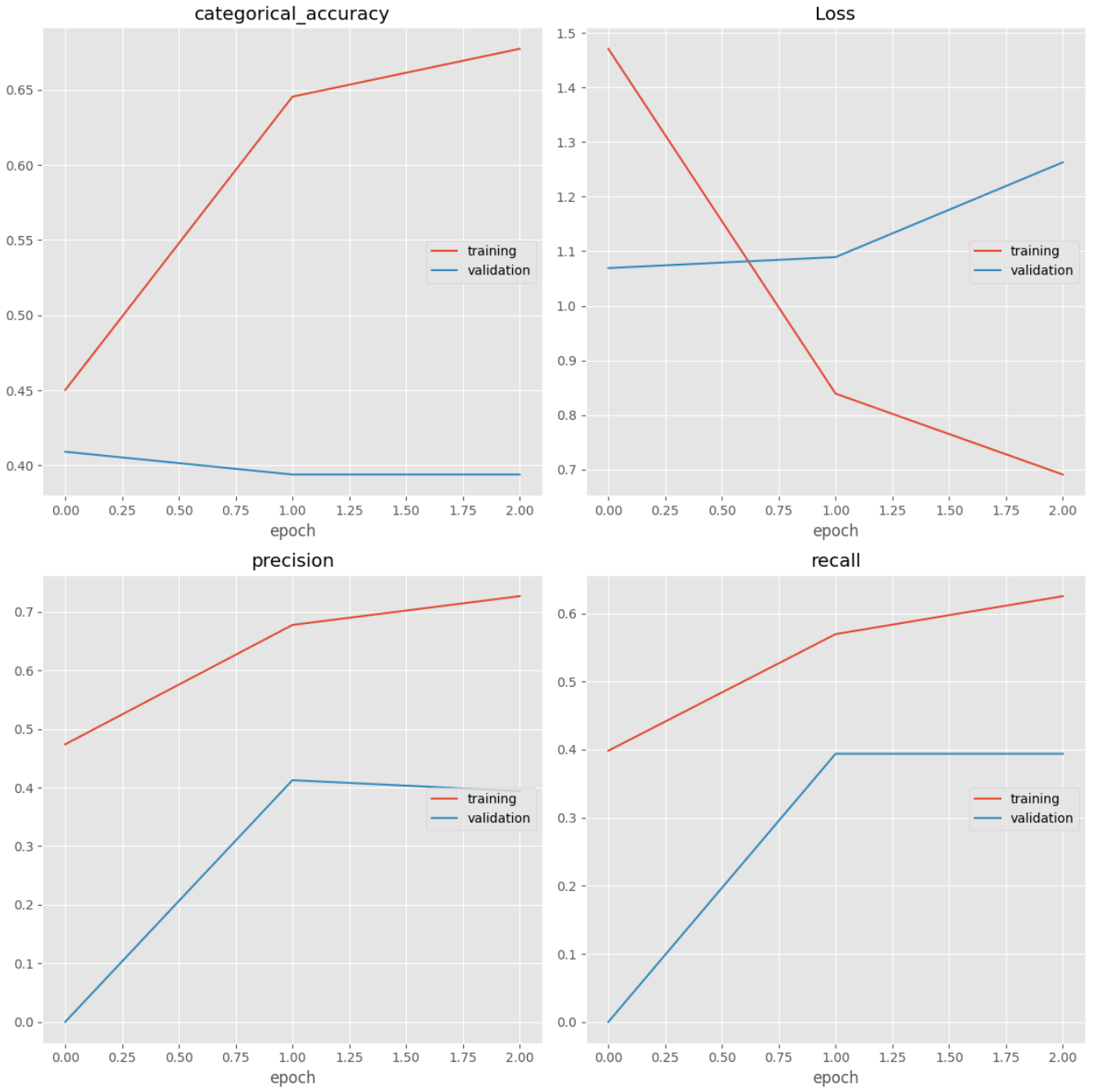
In [13]:
```
es = EarlyStopping(monitor='val_loss', mode='min', verbose=1, patience=2)
mc = ModelCheckpoint('./model_objects/initial_model.h5', monitor='val_loss', mode='mi
EPOCHS = 10
```

7. Train using a generator and test the accuracy on the test data at each epoch

8. Plot training & validation accuracy & loss

In [14]:
```
model.fit(
    train_generator,
```

```
    epochs=EPOCHS,
    validation_data=test_generator,
    callbacks=[
        es,
        mc,
        PlotLossesKerasTF()
    ]
)
```




```
categorical_accuracy
        training                (min:    0.450, max:    0.677, cur:    0.677)
        validation              (min:    0.394, max:    0.409, cur:    0.394)
Loss
        training                (min:    0.691, max:    1.471, cur:    0.691)
        validation              (min:    1.069, max:    1.263, cur:    1.263)
precision
        training                (min:    0.474, max:    0.727, cur:    0.727)
        validation              (min:    0.000, max:    0.413, cur:    0.394)
recall
        training                (min:    0.398, max:    0.625, cur:    0.625)
        validation              (min:    0.000, max:    0.394, cur:    0.394)
16/16 [==============================] - 12s 738ms/step - loss: 0.6907 - categorical_
accuracy: 0.6773 - precision: 0.7269 - recall: 0.6255 - val_loss: 1.2628 - val_catego
rical_accuracy: 0.3939 - val_precision: 0.3939 - val_recall: 0.3939
Epoch 3: early stopping
```

Out[14]: <keras.callbacks.History at 0x13759b760>

9. Observe Precision, Recall, F1-Score for all classes on both grayscale & color models - determine if the classes are good.

In [15]:
```
# Use sklearn classification report to evaluate the model
best_model = load_model('./model_objects/initial_model.h5')
y_pred = best_model.predict(test_generator)
y_pred = np.argmax(y_pred, axis=1)
y_true = test_generator.classes
print(classification_report(y_true, y_pred, target_names=classes))
print(f'Precision:\t{precision_score(y_true, y_pred, average="macro")}')
print(f'Recall:\t\t{recall_score(y_true, y_pred, average="macro")}')
print(f'F1 Score:\t{f1_score(y_true, y_pred, average="macro")}')

# The model is simply predicting the majority class for all images (i.e. everyone has
# The classes are quite bad
```

```
5/5 [==============================] - 1s 218ms/step
                precision    recall  f1-score   support

       Healthy       0.67      0.10      0.17        20
Type 1 disease       0.41      1.00      0.58        26
Type 2 disease       0.00      0.00      0.00        20

      accuracy                           0.42        66
     macro avg       0.36      0.37      0.25        66
  weighted avg       0.36      0.42      0.28        66

Precision:      0.35978835978835977
Recall:         0.3666666666666667
F1 Score:       0.25272756879986974
```

```
/usr/local/lib/python3.10/site-packages/sklearn/metrics/_classification.py:1334: UndefinedMetricWarning: Precision and F-score are ill-defined and being set to 0.0 in labels with no predicted samples. Use `zero_division` parameter to control this behavior.
  _warn_prf(average, modifier, msg_start, len(result))
/usr/local/lib/python3.10/site-packages/sklearn/metrics/_classification.py:1334: UndefinedMetricWarning: Precision and F-score are ill-defined and being set to 0.0 in labels with no predicted samples. Use `zero_division` parameter to control this behavior.
  _warn_prf(average, modifier, msg_start, len(result))
/usr/local/lib/python3.10/site-packages/sklearn/metrics/_classification.py:1334: UndefinedMetricWarning: Precision and F-score are ill-defined and being set to 0.0 in labels with no predicted samples. Use `zero_division` parameter to control this behavior.
  _warn_prf(average, modifier, msg_start, len(result))
/usr/local/lib/python3.10/site-packages/sklearn/metrics/_classification.py:1334: UndefinedMetricWarning: Precision is ill-defined and being set to 0.0 in labels with no predicted samples. Use `zero_division` parameter to control this behavior.
  _warn_prf(average, modifier, msg_start, len(result))
```

Section 3: Transfer Learning - Mobile Net

1. Prepare the dataset for the mobile-net model with color mode RGB

In [16]:
```
# Import MobileNetV2
from tensorflow.keras.applications import MobileNetV2
```

In [17]:
```
# Create an image generator for the training set & test set for the mobilenet model
mobilenet_train_generator = train_gen.flow_from_directory(
    train_path,
    target_size=(224,224),
```

```
    batch_size=16,
    class_mode='categorical',
    shuffle=True
)

mobilenet_test_generator = test_gen.flow_from_directory(
    test_path,
    target_size=(224,224),
    batch_size=16,
    class_mode='categorical',
    shuffle=True
)
```

```
Found 251 images belonging to 3 classes.
Found 66 images belonging to 3 classes.
```

2. Create an instance of the mobile-net pre-trained model

In [18]:
```
# Create the base model from the pre-trained model MobileNet V2
mobilenet_base_model = MobileNetV2(input_shape=(224,224,3), include_top=False, weight

# Freeze the base model
for layer in mobilenet_base_model.layers:
    layer.trainable = False
```

3. Add a dense layer, dropout layer, and batch normalization layer on the pre-trained model

In [19]:
```
# Create a new model on top with a dense layer, dropout layer, and batch normalizatio
x = BatchNormalization()(mobilenet_base_model.output)
x = MaxPooling2D((2,2))(x)
x = Dropout(0.2)(x)
x = Flatten()(x)
x = Dense(128, activation=relu)(x)
```

4. Create a final output using the softmax activation function

In [20]:
```
output_tensor = Dense(3, activation=softmax)(x)
mobilenet_model = Model(inputs=mobilenet_base_model.input, outputs=output_tensor)
```

5. Change the batch size activation function and optimize as rmsprop - observe if the accuracy increases

6. Change the loss function to categorical cross-entropy

In [21]:
```
mobilenet_model.compile(
    loss=categorical_crossentropy,
    optimizer=RMSprop(learning_rate=0.001),
    metrics=METRICS
)

mobilenet_model.summary()
```

```
Model: "model"
__________________________________________________________________________________________
 Layer (type)                Output Shape              Param #     Connected to
==========================================================================================
 input_2 (InputLayer)        [(None, 224, 224, 3       0           []
                             )]

 Conv1 (Conv2D)              (None, 112, 112, 32       864         ['input_2[0][0]']
                             )

 bn_Conv1 (BatchNormalization)  (None, 112, 112, 32    128         ['Conv1[0][0]']
                             )

 Conv1_relu (ReLU)           (None, 112, 112, 32       0           ['bn_Conv1[0][0]']
                             )

 expanded_conv_depthwise (Depth  (None, 112, 112, 32   288         ['Conv1_relu[0][0]']
 wiseConv2D)                 )

 expanded_conv_depthwise_BN (Ba  (None, 112, 112, 32   128         ['expanded_conv_dept
hwise[0][0]']
 tchNormalization)           )

 expanded_conv_depthwise_relu (  (None, 112, 112, 32   0           ['expanded_conv_dept
hwise_BN[0][0
 ReLU)                       )                                    ]']

 expanded_conv_project (Conv2D)  (None, 112, 112, 16   512         ['expanded_conv_dept
hwise_relu[0]
                             )                                    [0]']

 expanded_conv_project_BN (Batc  (None, 112, 112, 16   64          ['expanded_conv_proj
ect[0][0]']
 hNormalization)             )

 block_1_expand (Conv2D)     (None, 112, 112, 96   1536          ['expanded_conv_proj
ect_BN[0][0]'
                             )                                    ]

 block_1_expand_BN (BatchNormal  (None, 112, 112, 96   384       ['block_1_expand[0]
[0]']
 ization)                    )

 block_1_expand_relu (ReLU)  (None, 112, 112, 96   0           ['block_1_expand_BN
[0][0]']
                             )

 block_1_pad (ZeroPadding2D)  (None, 113, 113, 96   0          ['block_1_expand_rel
u[0][0]']
                             )

 block_1_depthwise (DepthwiseCo  (None, 56, 56, 96)  864       ['block_1_pad[0]
[0]']
 nv2D)

 block_1_depthwise_BN (BatchNor  (None, 56, 56, 96)  384       ['block_1_depthwise
[0][0]']
 malization)

 block_1_depthwise_relu (ReLU)  (None, 56, 56, 96)   0         ['block_1_depthwise_
BN[0][0]']

 block_1_project (Conv2D)    (None, 56, 56, 24)    2304        ['block_1_depthwise_
```

```
relu[0][0]']

 block_1_project_BN (BatchNorma  (None, 56, 56, 24)  96          ['block_1_project[0]
[0]']
 lization)

 block_2_expand (Conv2D)        (None, 56, 56, 144)  3456        ['block_1_project_BN
[0][0]']

 block_2_expand_BN (BatchNormal  (None, 56, 56, 144)  576        ['block_2_expand[0]
[0]']
 ization)

 block_2_expand_relu (ReLU)     (None, 56, 56, 144)  0           ['block_2_expand_BN
[0][0]']

 block_2_depthwise (DepthwiseCo  (None, 56, 56, 144)  1296       ['block_2_expand_rel
u[0][0]']
 nv2D)

 block_2_depthwise_BN (BatchNor  (None, 56, 56, 144)  576        ['block_2_depthwise
[0][0]']
 malization)

 block_2_depthwise_relu (ReLU)  (None, 56, 56, 144)  0           ['block_2_depthwise_
BN[0][0]']

 block_2_project (Conv2D)       (None, 56, 56, 24)   3456        ['block_2_depthwise_
relu[0][0]']

 block_2_project_BN (BatchNorma  (None, 56, 56, 24)  96          ['block_2_project[0]
[0]']
 lization)

 block_2_add (Add)              (None, 56, 56, 24)   0           ['block_1_project_BN
[0][0]',
                                                                  'block_2_project_BN
[0][0]']

 block_3_expand (Conv2D)        (None, 56, 56, 144)  3456        ['block_2_add[0]
[0]']

 block_3_expand_BN (BatchNormal  (None, 56, 56, 144)  576        ['block_3_expand[0]
[0]']
 ization)

 block_3_expand_relu (ReLU)     (None, 56, 56, 144)  0           ['block_3_expand_BN
[0][0]']

 block_3_pad (ZeroPadding2D)    (None, 57, 57, 144)  0           ['block_3_expand_rel
u[0][0]']

 block_3_depthwise (DepthwiseCo  (None, 28, 28, 144)  1296       ['block_3_pad[0]
[0]']
 nv2D)

 block_3_depthwise_BN (BatchNor  (None, 28, 28, 144)  576        ['block_3_depthwise
[0][0]']
 malization)

 block_3_depthwise_relu (ReLU)  (None, 28, 28, 144)  0           ['block_3_depthwise_
BN[0][0]']

 block_3_project (Conv2D)       (None, 28, 28, 32)   4608        ['block_3_depthwise_
relu[0][0]']
```

```
 block_3_project_BN (BatchNorma  (None, 28, 28, 32)  128         ['block_3_project[0]
[0]']
 lization)

 block_4_expand (Conv2D)        (None, 28, 28, 192)  6144        ['block_3_project_BN
[0][0]']

 block_4_expand_BN (BatchNormal  (None, 28, 28, 192)  768        ['block_4_expand[0]
[0]']
 ization)

 block_4_expand_relu (ReLU)     (None, 28, 28, 192)  0           ['block_4_expand_BN
[0][0]']

 block_4_depthwise (DepthwiseCo  (None, 28, 28, 192)  1728       ['block_4_expand_rel
u[0][0]']
 nv2D)

 block_4_depthwise_BN (BatchNor  (None, 28, 28, 192)  768        ['block_4_depthwise
[0][0]']
 malization)

 block_4_depthwise_relu (ReLU)  (None, 28, 28, 192)  0           ['block_4_depthwise_
BN[0][0]']

 block_4_project (Conv2D)       (None, 28, 28, 32)   6144        ['block_4_depthwise_
relu[0][0]']

 block_4_project_BN (BatchNorma  (None, 28, 28, 32)  128         ['block_4_project[0]
[0]']
 lization)

 block_4_add (Add)              (None, 28, 28, 32)   0           ['block_3_project_BN
[0][0]',
                                                                  'block_4_project_BN
[0][0]']

 block_5_expand (Conv2D)        (None, 28, 28, 192)  6144        ['block_4_add[0]
[0]']

 block_5_expand_BN (BatchNormal  (None, 28, 28, 192)  768        ['block_5_expand[0]
[0]']
 ization)

 block_5_expand_relu (ReLU)     (None, 28, 28, 192)  0           ['block_5_expand_BN
[0][0]']

 block_5_depthwise (DepthwiseCo  (None, 28, 28, 192)  1728       ['block_5_expand_rel
u[0][0]']
 nv2D)

 block_5_depthwise_BN (BatchNor  (None, 28, 28, 192)  768        ['block_5_depthwise
[0][0]']
 malization)

 block_5_depthwise_relu (ReLU)  (None, 28, 28, 192)  0           ['block_5_depthwise_
BN[0][0]']

 block_5_project (Conv2D)       (None, 28, 28, 32)   6144        ['block_5_depthwise_
relu[0][0]']

 block_5_project_BN (BatchNorma  (None, 28, 28, 32)  128         ['block_5_project[0]
[0]']
 lization)
```

```
 block_5_add (Add)              (None, 28, 28, 32)   0           ['block_4_add[0]
[0]',
                                                                  'block_5_project_BN
[0][0]']

 block_6_expand (Conv2D)        (None, 28, 28, 192)  6144        ['block_5_add[0]
[0]']

 block_6_expand_BN (BatchNormal  (None, 28, 28, 192)  768        ['block_6_expand[0]
[0]']
 ization)

 block_6_expand_relu (ReLU)     (None, 28, 28, 192)  0           ['block_6_expand_BN
[0][0]']

 block_6_pad (ZeroPadding2D)    (None, 29, 29, 192)  0           ['block_6_expand_rel
u[0][0]']

 block_6_depthwise (DepthwiseCo  (None, 14, 14, 192)  1728       ['block_6_pad[0]
[0]']
 nv2D)

 block_6_depthwise_BN (BatchNor  (None, 14, 14, 192)  768        ['block_6_depthwise
[0][0]']
 malization)

 block_6_depthwise_relu (ReLU)  (None, 14, 14, 192)  0           ['block_6_depthwise_
BN[0][0]']

 block_6_project (Conv2D)       (None, 14, 14, 64)   12288       ['block_6_depthwise_
relu[0][0]']

 block_6_project_BN (BatchNorma  (None, 14, 14, 64)  256         ['block_6_project[0]
[0]']
 lization)

 block_7_expand (Conv2D)        (None, 14, 14, 384)  24576       ['block_6_project_BN
[0][0]']

 block_7_expand_BN (BatchNormal  (None, 14, 14, 384)  1536       ['block_7_expand[0]
[0]']
 ization)

 block_7_expand_relu (ReLU)     (None, 14, 14, 384)  0           ['block_7_expand_BN
[0][0]']

 block_7_depthwise (DepthwiseCo  (None, 14, 14, 384)  3456       ['block_7_expand_rel
u[0][0]']
 nv2D)

 block_7_depthwise_BN (BatchNor  (None, 14, 14, 384)  1536       ['block_7_depthwise
[0][0]']
 malization)

 block_7_depthwise_relu (ReLU)  (None, 14, 14, 384)  0           ['block_7_depthwise_
BN[0][0]']

 block_7_project (Conv2D)       (None, 14, 14, 64)   24576       ['block_7_depthwise_
relu[0][0]']

 block_7_project_BN (BatchNorma  (None, 14, 14, 64)  256         ['block_7_project[0]
[0]']
 lization)
```

```
 block_7_add (Add)              (None, 14, 14, 64)   0           ['block_6_project_BN
[0][0]',
                                                                  'block_7_project_BN
[0][0]']

 block_8_expand (Conv2D)        (None, 14, 14, 384)  24576       ['block_7_add[0]
[0]']

 block_8_expand_BN (BatchNormal  (None, 14, 14, 384)  1536       ['block_8_expand[0]
[0]']
 ization)

 block_8_expand_relu (ReLU)     (None, 14, 14, 384)  0           ['block_8_expand_BN
[0][0]']

 block_8_depthwise (DepthwiseCo  (None, 14, 14, 384)  3456       ['block_8_expand_rel
u[0][0]']
 nv2D)

 block_8_depthwise_BN (BatchNor  (None, 14, 14, 384)  1536       ['block_8_depthwise
[0][0]']
 malization)

 block_8_depthwise_relu (ReLU)  (None, 14, 14, 384)  0           ['block_8_depthwise_
BN[0][0]']

 block_8_project (Conv2D)       (None, 14, 14, 64)   24576       ['block_8_depthwise_
relu[0][0]']

 block_8_project_BN (BatchNorma  (None, 14, 14, 64)  256         ['block_8_project[0]
[0]']
 lization)

 block_8_add (Add)              (None, 14, 14, 64)   0           ['block_7_add[0]
[0]',
                                                                  'block_8_project_BN
[0][0]']

 block_9_expand (Conv2D)        (None, 14, 14, 384)  24576       ['block_8_add[0]
[0]']

 block_9_expand_BN (BatchNormal  (None, 14, 14, 384)  1536       ['block_9_expand[0]
[0]']
 ization)

 block_9_expand_relu (ReLU)     (None, 14, 14, 384)  0           ['block_9_expand_BN
[0][0]']

 block_9_depthwise (DepthwiseCo  (None, 14, 14, 384)  3456       ['block_9_expand_rel
u[0][0]']
 nv2D)

 block_9_depthwise_BN (BatchNor  (None, 14, 14, 384)  1536       ['block_9_depthwise
[0][0]']
 malization)

 block_9_depthwise_relu (ReLU)  (None, 14, 14, 384)  0           ['block_9_depthwise_
BN[0][0]']

 block_9_project (Conv2D)       (None, 14, 14, 64)   24576       ['block_9_depthwise_
relu[0][0]']

 block_9_project_BN (BatchNorma  (None, 14, 14, 64)  256         ['block_9_project[0]
[0]']
 lization)
```

```
 block_9_add (Add)              (None, 14, 14, 64)   0           ['block_8_add[0]
[0]',
                                                                  'block_9_project_BN
[0][0]']

 block_10_expand (Conv2D)       (None, 14, 14, 384)  24576       ['block_9_add[0]
[0]']

 block_10_expand_BN (BatchNorma  (None, 14, 14, 384)  1536       ['block_10_expand[0]
[0]']
 lization)

 block_10_expand_relu (ReLU)    (None, 14, 14, 384)  0           ['block_10_expand_BN
[0][0]']

 block_10_depthwise (DepthwiseC  (None, 14, 14, 384)  3456       ['block_10_expand_re
lu[0][0]']
 onv2D)

 block_10_depthwise_BN (BatchNo  (None, 14, 14, 384)  1536       ['block_10_depthwise
[0][0]']
 rmalization)

 block_10_depthwise_relu (ReLU)  (None, 14, 14, 384)  0          ['block_10_depthwise
_BN[0][0]']

 block_10_project (Conv2D)      (None, 14, 14, 96)   36864       ['block_10_depthwise
_relu[0][0]']

 block_10_project_BN (BatchNorm  (None, 14, 14, 96)  384         ['block_10_project
[0][0]']
 alization)

 block_11_expand (Conv2D)       (None, 14, 14, 576)  55296       ['block_10_project_B
N[0][0]']

 block_11_expand_BN (BatchNorma  (None, 14, 14, 576)  2304       ['block_11_expand[0]
[0]']
 lization)

 block_11_expand_relu (ReLU)    (None, 14, 14, 576)  0           ['block_11_expand_BN
[0][0]']

 block_11_depthwise (DepthwiseC  (None, 14, 14, 576)  5184       ['block_11_expand_re
lu[0][0]']
 onv2D)

 block_11_depthwise_BN (BatchNo  (None, 14, 14, 576)  2304       ['block_11_depthwise
[0][0]']
 rmalization)

 block_11_depthwise_relu (ReLU)  (None, 14, 14, 576)  0          ['block_11_depthwise
_BN[0][0]']

 block_11_project (Conv2D)      (None, 14, 14, 96)   55296       ['block_11_depthwise
_relu[0][0]']

 block_11_project_BN (BatchNorm  (None, 14, 14, 96)  384         ['block_11_project
[0][0]']
 alization)

 block_11_add (Add)             (None, 14, 14, 96)   0           ['block_10_project_B
N[0][0]',
                                                                  'block_11_project_B
```

```
N[0][0]']

 block_12_expand (Conv2D)       (None, 14, 14, 576)  55296       ['block_11_add[0]
[0]']

 block_12_expand_BN (BatchNorma  (None, 14, 14, 576)  2304       ['block_12_expand[0]
[0]']
 lization)

 block_12_expand_relu (ReLU)    (None, 14, 14, 576)  0           ['block_12_expand_BN
[0][0]']

 block_12_depthwise (DepthwiseC  (None, 14, 14, 576)  5184       ['block_12_expand_re
lu[0][0]']
 onv2D)

 block_12_depthwise_BN (BatchNo  (None, 14, 14, 576)  2304       ['block_12_depthwise
[0][0]']
 rmalization)

 block_12_depthwise_relu (ReLU)  (None, 14, 14, 576)  0          ['block_12_depthwise
_BN[0][0]']

 block_12_project (Conv2D)      (None, 14, 14, 96)   55296       ['block_12_depthwise
_relu[0][0]']

 block_12_project_BN (BatchNorm  (None, 14, 14, 96)  384         ['block_12_project
[0][0]']
 alization)

 block_12_add (Add)             (None, 14, 14, 96)   0           ['block_11_add[0]
[0]',
                                                                  'block_12_project_B
N[0][0]']

 block_13_expand (Conv2D)       (None, 14, 14, 576)  55296       ['block_12_add[0]
[0]']

 block_13_expand_BN (BatchNorma  (None, 14, 14, 576)  2304       ['block_13_expand[0]
[0]']
 lization)

 block_13_expand_relu (ReLU)    (None, 14, 14, 576)  0           ['block_13_expand_BN
[0][0]']

 block_13_pad (ZeroPadding2D)   (None, 15, 15, 576)  0           ['block_13_expand_re
lu[0][0]']

 block_13_depthwise (DepthwiseC  (None, 7, 7, 576)   5184        ['block_13_pad[0]
[0]']
 onv2D)

 block_13_depthwise_BN (BatchNo  (None, 7, 7, 576)   2304        ['block_13_depthwise
[0][0]']
 rmalization)

 block_13_depthwise_relu (ReLU)  (None, 7, 7, 576)   0          ['block_13_depthwise
_BN[0][0]']

 block_13_project (Conv2D)      (None, 7, 7, 160)    92160       ['block_13_depthwise
_relu[0][0]']

 block_13_project_BN (BatchNorm  (None, 7, 7, 160)   640         ['block_13_project
[0][0]']
 alization)
```

```
 block_14_expand (Conv2D)       (None, 7, 7, 960)    153600      ['block_13_project_B
N[0][0]']

 block_14_expand_BN (BatchNorma  (None, 7, 7, 960)   3840        ['block_14_expand[0]
[0]']
 lization)

 block_14_expand_relu (ReLU)    (None, 7, 7, 960)    0           ['block_14_expand_BN
[0][0]']

 block_14_depthwise (DepthwiseC  (None, 7, 7, 960)   8640        ['block_14_expand_re
lu[0][0]']
 onv2D)

 block_14_depthwise_BN (BatchNo  (None, 7, 7, 960)   3840        ['block_14_depthwise
[0][0]']
 rmalization)

 block_14_depthwise_relu (ReLU)  (None, 7, 7, 960)   0           ['block_14_depthwise
_BN[0][0]']

 block_14_project (Conv2D)      (None, 7, 7, 160)    153600      ['block_14_depthwise
_relu[0][0]']

 block_14_project_BN (BatchNorm  (None, 7, 7, 160)   640         ['block_14_project
[0][0]']
 alization)

 block_14_add (Add)             (None, 7, 7, 160)    0           ['block_13_project_B
N[0][0]',
                                                                  'block_14_project_B
N[0][0]']

 block_15_expand (Conv2D)       (None, 7, 7, 960)    153600      ['block_14_add[0]
[0]']

 block_15_expand_BN (BatchNorma  (None, 7, 7, 960)   3840        ['block_15_expand[0]
[0]']
 lization)

 block_15_expand_relu (ReLU)    (None, 7, 7, 960)    0           ['block_15_expand_BN
[0][0]']

 block_15_depthwise (DepthwiseC  (None, 7, 7, 960)   8640        ['block_15_expand_re
lu[0][0]']
 onv2D)

 block_15_depthwise_BN (BatchNo  (None, 7, 7, 960)   3840        ['block_15_depthwise
[0][0]']
 rmalization)

 block_15_depthwise_relu (ReLU)  (None, 7, 7, 960)   0           ['block_15_depthwise
_BN[0][0]']

 block_15_project (Conv2D)      (None, 7, 7, 160)    153600      ['block_15_depthwise
_relu[0][0]']

 block_15_project_BN (BatchNorm  (None, 7, 7, 160)   640         ['block_15_project
[0][0]']
 alization)

 block_15_add (Add)             (None, 7, 7, 160)    0           ['block_14_add[0]
[0]',
                                                                  'block_15_project_B
```

```
N[0][0]']

 block_16_expand (Conv2D)       (None, 7, 7, 960)    153600      ['block_15_add[0]
[0]']

 block_16_expand_BN (BatchNorma  (None, 7, 7, 960)   3840        ['block_16_expand[0]
[0]']
 lization)

 block_16_expand_relu (ReLU)    (None, 7, 7, 960)    0           ['block_16_expand_BN
[0][0]']

 block_16_depthwise (DepthwiseC  (None, 7, 7, 960)   8640        ['block_16_expand_re
lu[0][0]']
 onv2D)

 block_16_depthwise_BN (BatchNo  (None, 7, 7, 960)   3840        ['block_16_depthwise
[0][0]']
 rmalization)

 block_16_depthwise_relu (ReLU)  (None, 7, 7, 960)   0           ['block_16_depthwise
_BN[0][0]']

 block_16_project (Conv2D)      (None, 7, 7, 320)    307200      ['block_16_depthwise
_relu[0][0]']

 block_16_project_BN (BatchNorm  (None, 7, 7, 320)   1280        ['block_16_project
[0][0]']
 alization)

 Conv_1 (Conv2D)                (None, 7, 7, 1280)   409600      ['block_16_project_B
N[0][0]']

 Conv_1_bn (BatchNormalization)  (None, 7, 7, 1280)  5120        ['Conv_1[0][0]']

 out_relu (ReLU)                (None, 7, 7, 1280)   0           ['Conv_1_bn[0][0]']

 batch_normalization_3 (BatchNo  (None, 7, 7, 1280)  5120        ['out_relu[0][0]']
 rmalization)

 max_pooling2d_3 (MaxPooling2D)  (None, 3, 3, 1280)  0           ['batch_normalizatio
n_3[0][0]']

 dropout_5 (Dropout)            (None, 3, 3, 1280)   0           ['max_pooling2d_3[0]
[0]']

 flatten_1 (Flatten)            (None, 11520)        0           ['dropout_5[0][0]']

 dense_3 (Dense)                (None, 128)          1474688     ['flatten_1[0][0]']

 dense_4 (Dense)                (None, 3)            387         ['dense_3[0][0]']

==================================================================================
=============
Total params: 3,738,179
Trainable params: 1,477,635
Non-trainable params: 2,260,544
__________________________________________________________________________________
_____________
```

7. Use an early stopping callback on the validation loss with a patience of 2 epochs to prevent overfitting

8. Use 10 epochs

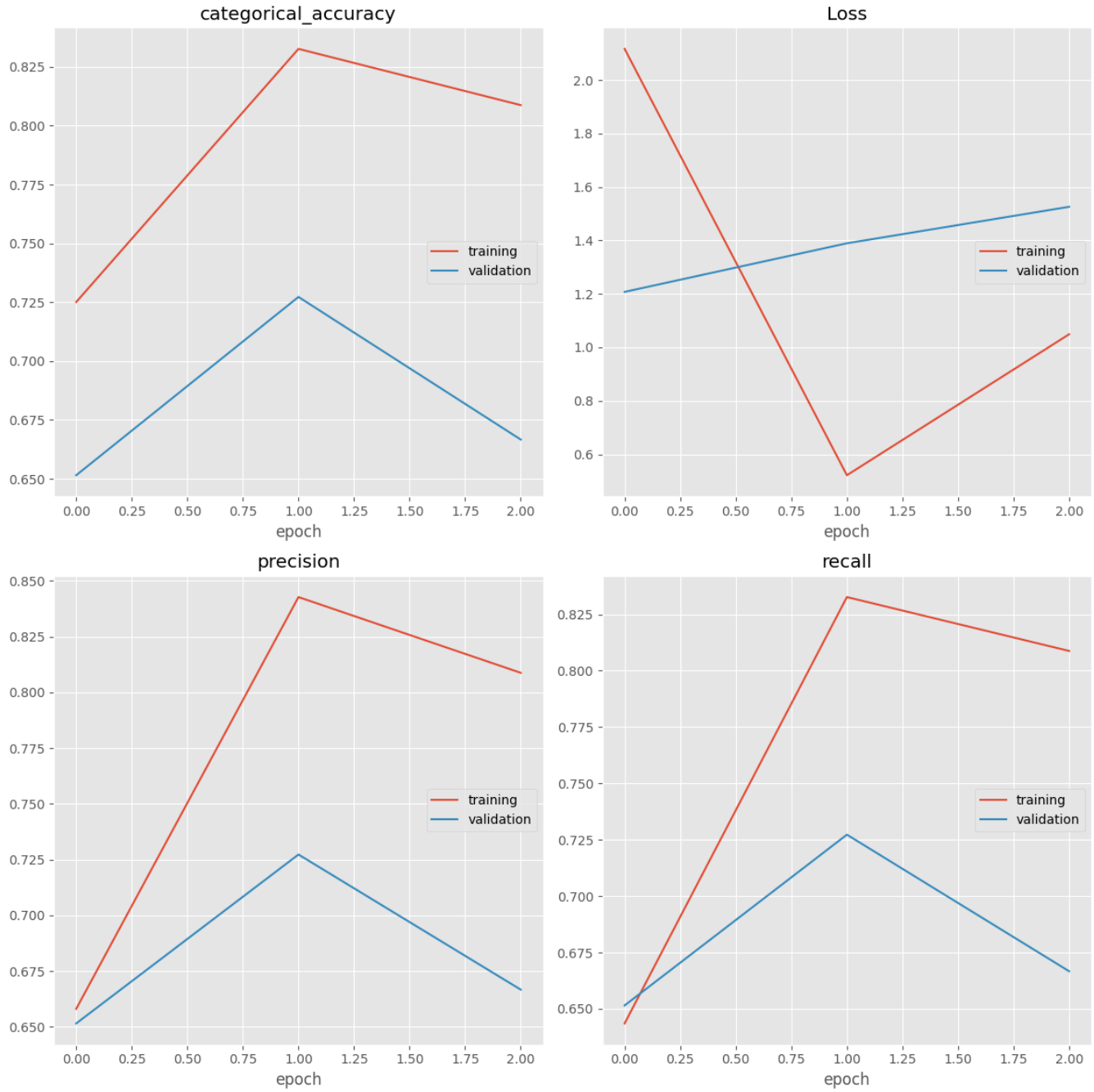
In [22]:
```
es = EarlyStopping(monitor='val_loss', mode='min', verbose=1, patience=2)
```

```
mc = ModelCheckpoint('./model_objects/mobilenet_model.h5', monitor='val_loss', mode='
EPOCHS = 10
```

9. Train using a generator and test the accuracy on the test data at each epoch

10. Plot training & validation accuracy & loss

In [23]:
```
mobilenet_model.fit(
    mobilenet_train_generator,
    epochs=EPOCHS,
    validation_data=mobilenet_test_generator,
    callbacks=[
        es,
        mc,
        PlotLossesKerasTF()
    ]
)
```



```
categorical_accuracy
        training                (min:    0.725, max:    0.833, cur:    0.809)
        validation              (min:    0.652, max:    0.727, cur:    0.667)
Loss
        training                (min:    0.522, max:    2.117, cur:    1.049)
        validation              (min:    1.208, max:    1.526, cur:    1.526)
precision
        training                (min:    0.658, max:    0.843, cur:    0.809)
        validation              (min:    0.652, max:    0.727, cur:    0.667)
recall
        training                (min:    0.644, max:    0.833, cur:    0.809)
        validation              (min:    0.652, max:    0.727, cur:    0.667)
16/16 [==============================] - 15s 967ms/step - loss: 1.0490 - categorical_
accuracy: 0.8088 - precision: 0.8088 - recall: 0.8088 - val_loss: 1.5258 - val_catego
rical_accuracy: 0.6667 - val_precision: 0.6667 - val_recall: 0.6667
Epoch 3: early stopping
```

Out[23]:
```
<keras.callbacks.History at 0x138374b20>
```

11. Observe Precision, Recall, F1-Score for all classes on both grayscale & color models - determine if the classes are good.

In [24]:
```
# Use sklearn classification report to evaluate the mobilenet model
best_mobilenet_model = load_model('./model_objects/mobilenet_model.h5')
y_pred = best_mobilenet_model.predict(mobilenet_test_generator)
y_pred = np.argmax(y_pred, axis=1)
y_true = mobilenet_test_generator.classes
print(classification_report(y_true, y_pred, target_names=classes))
print(f'Precision:\t{precision_score(y_true, y_pred, average="macro")}')
print(f'Recall:\t\t{recall_score(y_true, y_pred, average="macro")}')
print(f'F1 Score:\t{f1_score(y_true, y_pred, average="macro")}')

# F1 score is rather low still for the mobilenet model.
# This comes despite rather high categorical accuracy (0.924)
# We are predicting disease so we should expect a lower F1 Score - but Recall & Preci
# We should expect better precision
```

```
5/5 [==============================] - 3s 323ms/step
                precision    recall  f1-score   support

       Healthy       0.33      0.05      0.09        20
Type 1 disease       0.40      0.31      0.35        26
Type 2 disease       0.30      0.65      0.41        20

      accuracy                           0.33        66
     macro avg       0.35      0.34      0.28        66
  weighted avg       0.35      0.33      0.29        66

Precision:      0.3452196382428941
Recall:         0.33589743589743587
F1 Score:       0.28249367379802165
```

Section 4: Transfer Learning - Densenet121

1. Prepare the dataset for the densenet121 model with image size 224x224x3

In [25]:
```
# Import denseNet121
from tensorflow.keras.applications import DenseNet121
```

In [26]:
```
# Create an image generator for the training set & test set for the densenet model
densenet_train_generator = train_gen.flow_from_directory(
    train_path,
    target_size=(224,224),
    batch_size=16,
    class_mode='categorical',
```

```
    shuffle=True
)

densenet_test_generator = test_gen.flow_from_directory(
    test_path,
    target_size=(224,224),
    batch_size=16,
    class_mode='categorical',
    shuffle=True
)
```

```
Found 251 images belonging to 3 classes.
Found 66 images belonging to 3 classes.
```

In [27]:
```
# Create an instance of the DenseNet121 model
densenet_base_model = DenseNet121(input_shape=(224,224,3), include_top=False, weights
```

2. Freeze the top layers of the pre-trained model

In [28]:
```
# Loop over the layers in the base model and freeze them
for layer in densenet_base_model.layers:
    layer.trainable = False
```

3. Add a dense layer at the end of the pre-trained model, followed by a dropout layer and try various combinations to optimize accuracy

In [29]:
```
# Add a dense layer followed by a dropout layer on top of the base model
x = Dropout(0.2)(densenet_base_model.output)
x = Flatten()(x)
x = Dense(128, activation=relu)(x)
```

4. Create a final output using the softmax activation function

In [30]:
```
# Add the output layer and build the model
output_tensor = Dense(3, activation=softmax)(x)
densenet_model = Model(inputs=densenet_base_model.input, outputs=output_tensor)
```

5. Change the loss function to categorical cross-entropy

6. Use Adam as the optimizer

In [31]:
```
densenet_model.compile(
    loss=categorical_crossentropy,
    optimizer=Adam(learning_rate=0.001),
    metrics=METRICS
)

densenet_model.summary()
```

```
Model: "model_1"
__________________________________________________________________________________________
 Layer (type)                Output Shape              Param #     Connected to
==========================================================================================
 input_3 (InputLayer)        [(None, 224, 224, 3       0           []
                             )]

 zero_padding2d (ZeroPadding2D)  (None, 230, 230, 3)   0           ['input_3[0][0]']

 conv1/conv (Conv2D)         (None, 112, 112, 64       9408        ['zero_padding2d[0]
[0]']
                             )

 conv1/bn (BatchNormalization)  (None, 112, 112, 64    256         ['conv1/conv[0][0]']
                             )

 conv1/relu (Activation)     (None, 112, 112, 64       0           ['conv1/bn[0][0]']
                             )

 zero_padding2d_1 (ZeroPadding2  (None, 114, 114, 64   0          ['conv1/relu[0][0]']
 D)                          )

 pool1 (MaxPooling2D)        (None, 56, 56, 64)        0           ['zero_padding2d_1
[0][0]']

 conv2_block1_0_bn (BatchNormal  (None, 56, 56, 64)    256         ['pool1[0][0]']
 ization)

 conv2_block1_0_relu (Activatio  (None, 56, 56, 64)    0           ['conv2_block1_0_bn
[0][0]']
 n)

 conv2_block1_1_conv (Conv2D)   (None, 56, 56, 128)    8192        ['conv2_block1_0_rel
u[0][0]']

 conv2_block1_1_bn (BatchNormal  (None, 56, 56, 128)   512         ['conv2_block1_1_con
v[0][0]']
 ization)

 conv2_block1_1_relu (Activatio  (None, 56, 56, 128)   0           ['conv2_block1_1_bn
[0][0]']
 n)

 conv2_block1_2_conv (Conv2D)   (None, 56, 56, 32)     36864       ['conv2_block1_1_rel
u[0][0]']

 conv2_block1_concat (Concatena  (None, 56, 56, 96)    0           ['pool1[0][0]',
 te)                                                     'conv2_block1_2_con
v[0][0]']

 conv2_block2_0_bn (BatchNormal  (None, 56, 56, 96)    384         ['conv2_block1_conca
t[0][0]']
 ization)

 conv2_block2_0_relu (Activatio  (None, 56, 56, 96)    0           ['conv2_block2_0_bn
[0][0]']
 n)

 conv2_block2_1_conv (Conv2D)   (None, 56, 56, 128)    12288       ['conv2_block2_0_rel
u[0][0]']

 conv2_block2_1_bn (BatchNormal  (None, 56, 56, 128)   512         ['conv2_block2_1_con
v[0][0]']
```

```
 ization)

 conv2_block2_1_relu (Activatio  (None, 56, 56, 128)  0          ['conv2_block2_1_bn
[0][0]']
 n)

 conv2_block2_2_conv (Conv2D)   (None, 56, 56, 32)   36864       ['conv2_block2_1_rel
u[0][0]']

 conv2_block2_concat (Concatena  (None, 56, 56, 128)  0          ['conv2_block1_conca
t[0][0]',
 te)                                                             'conv2_block2_2_con
v[0][0]']

 conv2_block3_0_bn (BatchNormal  (None, 56, 56, 128)  512        ['conv2_block2_conca
t[0][0]']
 ization)

 conv2_block3_0_relu (Activatio  (None, 56, 56, 128)  0          ['conv2_block3_0_bn
[0][0]']
 n)

 conv2_block3_1_conv (Conv2D)   (None, 56, 56, 128)  16384       ['conv2_block3_0_rel
u[0][0]']

 conv2_block3_1_bn (BatchNormal  (None, 56, 56, 128)  512        ['conv2_block3_1_con
v[0][0]']
 ization)

 conv2_block3_1_relu (Activatio  (None, 56, 56, 128)  0          ['conv2_block3_1_bn
[0][0]']
 n)

 conv2_block3_2_conv (Conv2D)   (None, 56, 56, 32)   36864       ['conv2_block3_1_rel
u[0][0]']

 conv2_block3_concat (Concatena  (None, 56, 56, 160)  0          ['conv2_block2_conca
t[0][0]',
 te)                                                             'conv2_block3_2_con
v[0][0]']

 conv2_block4_0_bn (BatchNormal  (None, 56, 56, 160)  640        ['conv2_block3_conca
t[0][0]']
 ization)

 conv2_block4_0_relu (Activatio  (None, 56, 56, 160)  0          ['conv2_block4_0_bn
[0][0]']
 n)

 conv2_block4_1_conv (Conv2D)   (None, 56, 56, 128)  20480       ['conv2_block4_0_rel
u[0][0]']

 conv2_block4_1_bn (BatchNormal  (None, 56, 56, 128)  512        ['conv2_block4_1_con
v[0][0]']
 ization)

 conv2_block4_1_relu (Activatio  (None, 56, 56, 128)  0          ['conv2_block4_1_bn
[0][0]']
 n)

 conv2_block4_2_conv (Conv2D)   (None, 56, 56, 32)   36864       ['conv2_block4_1_rel
u[0][0]']

 conv2_block4_concat (Concatena  (None, 56, 56, 192)  0          ['conv2_block3_conca
t[0][0]',
```

```
 te)                                                              'conv2_block4_2_con
v[0][0]']

 conv2_block5_0_bn (BatchNormal  (None, 56, 56, 192)  768        ['conv2_block4_conca
t[0][0]']
 ization)

 conv2_block5_0_relu (Activatio  (None, 56, 56, 192)  0          ['conv2_block5_0_bn
[0][0]']
 n)

 conv2_block5_1_conv (Conv2D)   (None, 56, 56, 128)  24576       ['conv2_block5_0_rel
u[0][0]']

 conv2_block5_1_bn (BatchNormal  (None, 56, 56, 128)  512        ['conv2_block5_1_con
v[0][0]']
 ization)

 conv2_block5_1_relu (Activatio  (None, 56, 56, 128)  0          ['conv2_block5_1_bn
[0][0]']
 n)

 conv2_block5_2_conv (Conv2D)   (None, 56, 56, 32)   36864       ['conv2_block5_1_rel
u[0][0]']

 conv2_block5_concat (Concatena  (None, 56, 56, 224)  0          ['conv2_block4_conca
t[0][0]',
 te)                                                              'conv2_block5_2_con
v[0][0]']

 conv2_block6_0_bn (BatchNormal  (None, 56, 56, 224)  896        ['conv2_block5_conca
t[0][0]']
 ization)

 conv2_block6_0_relu (Activatio  (None, 56, 56, 224)  0          ['conv2_block6_0_bn
[0][0]']
 n)

 conv2_block6_1_conv (Conv2D)   (None, 56, 56, 128)  28672       ['conv2_block6_0_rel
u[0][0]']

 conv2_block6_1_bn (BatchNormal  (None, 56, 56, 128)  512        ['conv2_block6_1_con
v[0][0]']
 ization)

 conv2_block6_1_relu (Activatio  (None, 56, 56, 128)  0          ['conv2_block6_1_bn
[0][0]']
 n)

 conv2_block6_2_conv (Conv2D)   (None, 56, 56, 32)   36864       ['conv2_block6_1_rel
u[0][0]']

 conv2_block6_concat (Concatena  (None, 56, 56, 256)  0          ['conv2_block5_conca
t[0][0]',
 te)                                                              'conv2_block6_2_con
v[0][0]']

 pool2_bn (BatchNormalization)  (None, 56, 56, 256)  1024        ['conv2_block6_conca
t[0][0]']

 pool2_relu (Activation)        (None, 56, 56, 256)  0           ['pool2_bn[0][0]']

 pool2_conv (Conv2D)            (None, 56, 56, 128)  32768       ['pool2_relu[0][0]']

 pool2_pool (AveragePooling2D)  (None, 28, 28, 128)  0           ['pool2_conv[0][0]']
```

```
 conv3_block1_0_bn (BatchNormal  (None, 28, 28, 128)  512         ['pool2_pool[0][0]']
 ization)

 conv3_block1_0_relu (Activatio  (None, 28, 28, 128)  0           ['conv3_block1_0_bn
[0][0]']
 n)

 conv3_block1_1_conv (Conv2D)   (None, 28, 28, 128)  16384       ['conv3_block1_0_rel
u[0][0]']

 conv3_block1_1_bn (BatchNormal  (None, 28, 28, 128)  512         ['conv3_block1_1_con
v[0][0]']
 ization)

 conv3_block1_1_relu (Activatio  (None, 28, 28, 128)  0           ['conv3_block1_1_bn
[0][0]']
 n)

 conv3_block1_2_conv (Conv2D)   (None, 28, 28, 32)   36864       ['conv3_block1_1_rel
u[0][0]']

 conv3_block1_concat (Concatena  (None, 28, 28, 160)  0           ['pool2_pool[0][0]',
 te)                                                              'conv3_block1_2_con
v[0][0]']

 conv3_block2_0_bn (BatchNormal  (None, 28, 28, 160)  640         ['conv3_block1_conca
t[0][0]']
 ization)

 conv3_block2_0_relu (Activatio  (None, 28, 28, 160)  0           ['conv3_block2_0_bn
[0][0]']
 n)

 conv3_block2_1_conv (Conv2D)   (None, 28, 28, 128)  20480       ['conv3_block2_0_rel
u[0][0]']

 conv3_block2_1_bn (BatchNormal  (None, 28, 28, 128)  512         ['conv3_block2_1_con
v[0][0]']
 ization)

 conv3_block2_1_relu (Activatio  (None, 28, 28, 128)  0           ['conv3_block2_1_bn
[0][0]']
 n)

 conv3_block2_2_conv (Conv2D)   (None, 28, 28, 32)   36864       ['conv3_block2_1_rel
u[0][0]']

 conv3_block2_concat (Concatena  (None, 28, 28, 192)  0           ['conv3_block1_conca
t[0][0]',
 te)                                                              'conv3_block2_2_con
v[0][0]']

 conv3_block3_0_bn (BatchNormal  (None, 28, 28, 192)  768         ['conv3_block2_conca
t[0][0]']
 ization)

 conv3_block3_0_relu (Activatio  (None, 28, 28, 192)  0           ['conv3_block3_0_bn
[0][0]']
 n)

 conv3_block3_1_conv (Conv2D)   (None, 28, 28, 128)  24576       ['conv3_block3_0_rel
u[0][0]']

 conv3_block3_1_bn (BatchNormal  (None, 28, 28, 128)  512         ['conv3_block3_1_con
```

```
v[0][0]']
 ization)

 conv3_block3_1_relu (Activatio  (None, 28, 28, 128)  0          ['conv3_block3_1_bn
[0][0]']
 n)

 conv3_block3_2_conv (Conv2D)   (None, 28, 28, 32)   36864       ['conv3_block3_1_rel
u[0][0]']

 conv3_block3_concat (Concatena  (None, 28, 28, 224)  0          ['conv3_block2_conca
t[0][0]',
 te)                                                             'conv3_block3_2_con
v[0][0]']

 conv3_block4_0_bn (BatchNormal  (None, 28, 28, 224)  896        ['conv3_block3_conca
t[0][0]']
 ization)

 conv3_block4_0_relu (Activatio  (None, 28, 28, 224)  0          ['conv3_block4_0_bn
[0][0]']
 n)

 conv3_block4_1_conv (Conv2D)   (None, 28, 28, 128)  28672       ['conv3_block4_0_rel
u[0][0]']

 conv3_block4_1_bn (BatchNormal  (None, 28, 28, 128)  512        ['conv3_block4_1_con
v[0][0]']
 ization)

 conv3_block4_1_relu (Activatio  (None, 28, 28, 128)  0          ['conv3_block4_1_bn
[0][0]']
 n)

 conv3_block4_2_conv (Conv2D)   (None, 28, 28, 32)   36864       ['conv3_block4_1_rel
u[0][0]']

 conv3_block4_concat (Concatena  (None, 28, 28, 256)  0          ['conv3_block3_conca
t[0][0]',
 te)                                                             'conv3_block4_2_con
v[0][0]']

 conv3_block5_0_bn (BatchNormal  (None, 28, 28, 256)  1024       ['conv3_block4_conca
t[0][0]']
 ization)

 conv3_block5_0_relu (Activatio  (None, 28, 28, 256)  0          ['conv3_block5_0_bn
[0][0]']
 n)

 conv3_block5_1_conv (Conv2D)   (None, 28, 28, 128)  32768       ['conv3_block5_0_rel
u[0][0]']

 conv3_block5_1_bn (BatchNormal  (None, 28, 28, 128)  512        ['conv3_block5_1_con
v[0][0]']
 ization)

 conv3_block5_1_relu (Activatio  (None, 28, 28, 128)  0          ['conv3_block5_1_bn
[0][0]']
 n)

 conv3_block5_2_conv (Conv2D)   (None, 28, 28, 32)   36864       ['conv3_block5_1_rel
u[0][0]']

 conv3_block5_concat (Concatena  (None, 28, 28, 288)  0          ['conv3_block4_conca
```

```
t[0][0]',
 te)                                                                 'conv3_block5_2_con
v[0][0]']

 conv3_block6_0_bn (BatchNormal  (None, 28, 28, 288)  1152        ['conv3_block5_conca
t[0][0]']
 ization)

 conv3_block6_0_relu (Activatio  (None, 28, 28, 288)  0           ['conv3_block6_0_bn
[0][0]']
 n)

 conv3_block6_1_conv (Conv2D)   (None, 28, 28, 128)  36864       ['conv3_block6_0_rel
u[0][0]']

 conv3_block6_1_bn (BatchNormal  (None, 28, 28, 128)  512         ['conv3_block6_1_con
v[0][0]']
 ization)

 conv3_block6_1_relu (Activatio  (None, 28, 28, 128)  0           ['conv3_block6_1_bn
[0][0]']
 n)

 conv3_block6_2_conv (Conv2D)   (None, 28, 28, 32)   36864       ['conv3_block6_1_rel
u[0][0]']

 conv3_block6_concat (Concatena  (None, 28, 28, 320)  0           ['conv3_block5_conca
t[0][0]',
 te)                                                                 'conv3_block6_2_con
v[0][0]']

 conv3_block7_0_bn (BatchNormal  (None, 28, 28, 320)  1280        ['conv3_block6_conca
t[0][0]']
 ization)

 conv3_block7_0_relu (Activatio  (None, 28, 28, 320)  0           ['conv3_block7_0_bn
[0][0]']
 n)

 conv3_block7_1_conv (Conv2D)   (None, 28, 28, 128)  40960       ['conv3_block7_0_rel
u[0][0]']

 conv3_block7_1_bn (BatchNormal  (None, 28, 28, 128)  512         ['conv3_block7_1_con
v[0][0]']
 ization)

 conv3_block7_1_relu (Activatio  (None, 28, 28, 128)  0           ['conv3_block7_1_bn
[0][0]']
 n)

 conv3_block7_2_conv (Conv2D)   (None, 28, 28, 32)   36864       ['conv3_block7_1_rel
u[0][0]']

 conv3_block7_concat (Concatena  (None, 28, 28, 352)  0           ['conv3_block6_conca
t[0][0]',
 te)                                                                 'conv3_block7_2_con
v[0][0]']

 conv3_block8_0_bn (BatchNormal  (None, 28, 28, 352)  1408        ['conv3_block7_conca
t[0][0]']
 ization)

 conv3_block8_0_relu (Activatio  (None, 28, 28, 352)  0           ['conv3_block8_0_bn
[0][0]']
 n)
```

```
 conv3_block8_1_conv (Conv2D)   (None, 28, 28, 128)  45056       ['conv3_block8_0_rel
u[0][0]']

 conv3_block8_1_bn (BatchNormal  (None, 28, 28, 128)  512        ['conv3_block8_1_con
v[0][0]']
 ization)

 conv3_block8_1_relu (Activatio  (None, 28, 28, 128)  0          ['conv3_block8_1_bn
[0][0]']
 n)

 conv3_block8_2_conv (Conv2D)   (None, 28, 28, 32)   36864       ['conv3_block8_1_rel
u[0][0]']

 conv3_block8_concat (Concatena  (None, 28, 28, 384)  0          ['conv3_block7_conca
t[0][0]',
 te)                                                             'conv3_block8_2_con
v[0][0]']

 conv3_block9_0_bn (BatchNormal  (None, 28, 28, 384)  1536       ['conv3_block8_conca
t[0][0]']
 ization)

 conv3_block9_0_relu (Activatio  (None, 28, 28, 384)  0          ['conv3_block9_0_bn
[0][0]']
 n)

 conv3_block9_1_conv (Conv2D)   (None, 28, 28, 128)  49152       ['conv3_block9_0_rel
u[0][0]']

 conv3_block9_1_bn (BatchNormal  (None, 28, 28, 128)  512        ['conv3_block9_1_con
v[0][0]']
 ization)

 conv3_block9_1_relu (Activatio  (None, 28, 28, 128)  0          ['conv3_block9_1_bn
[0][0]']
 n)

 conv3_block9_2_conv (Conv2D)   (None, 28, 28, 32)   36864       ['conv3_block9_1_rel
u[0][0]']

 conv3_block9_concat (Concatena  (None, 28, 28, 416)  0          ['conv3_block8_conca
t[0][0]',
 te)                                                             'conv3_block9_2_con
v[0][0]']

 conv3_block10_0_bn (BatchNorma  (None, 28, 28, 416)  1664       ['conv3_block9_conca
t[0][0]']
 lization)

 conv3_block10_0_relu (Activati  (None, 28, 28, 416)  0          ['conv3_block10_0_bn
[0][0]']
 on)

 conv3_block10_1_conv (Conv2D)  (None, 28, 28, 128)  53248       ['conv3_block10_0_re
lu[0][0]']

 conv3_block10_1_bn (BatchNorma  (None, 28, 28, 128)  512        ['conv3_block10_1_co
nv[0][0]']
 lization)

 conv3_block10_1_relu (Activati  (None, 28, 28, 128)  0          ['conv3_block10_1_bn
[0][0]']
 on)
```

```
 conv3_block10_2_conv (Conv2D)  (None, 28, 28, 32)   36864       ['conv3_block10_1_re
lu[0][0]']

 conv3_block10_concat (Concaten  (None, 28, 28, 448)  0          ['conv3_block9_conca
t[0][0]',
 ate)                                                             'conv3_block10_2_co
nv[0][0]']

 conv3_block11_0_bn (BatchNorma  (None, 28, 28, 448)  1792       ['conv3_block10_conc
at[0][0]']
 lization)

 conv3_block11_0_relu (Activati  (None, 28, 28, 448)  0          ['conv3_block11_0_bn
[0][0]']
 on)

 conv3_block11_1_conv (Conv2D)  (None, 28, 28, 128)  57344       ['conv3_block11_0_re
lu[0][0]']

 conv3_block11_1_bn (BatchNorma  (None, 28, 28, 128)  512        ['conv3_block11_1_co
nv[0][0]']
 lization)

 conv3_block11_1_relu (Activati  (None, 28, 28, 128)  0          ['conv3_block11_1_bn
[0][0]']
 on)

 conv3_block11_2_conv (Conv2D)  (None, 28, 28, 32)   36864       ['conv3_block11_1_re
lu[0][0]']

 conv3_block11_concat (Concaten  (None, 28, 28, 480)  0          ['conv3_block10_conc
at[0][0]',
 ate)                                                             'conv3_block11_2_co
nv[0][0]']

 conv3_block12_0_bn (BatchNorma  (None, 28, 28, 480)  1920       ['conv3_block11_conc
at[0][0]']
 lization)

 conv3_block12_0_relu (Activati  (None, 28, 28, 480)  0          ['conv3_block12_0_bn
[0][0]']
 on)

 conv3_block12_1_conv (Conv2D)  (None, 28, 28, 128)  61440       ['conv3_block12_0_re
lu[0][0]']

 conv3_block12_1_bn (BatchNorma  (None, 28, 28, 128)  512        ['conv3_block12_1_co
nv[0][0]']
 lization)

 conv3_block12_1_relu (Activati  (None, 28, 28, 128)  0          ['conv3_block12_1_bn
[0][0]']
 on)

 conv3_block12_2_conv (Conv2D)  (None, 28, 28, 32)   36864       ['conv3_block12_1_re
lu[0][0]']

 conv3_block12_concat (Concaten  (None, 28, 28, 512)  0          ['conv3_block11_conc
at[0][0]',
 ate)                                                             'conv3_block12_2_co
nv[0][0]']

 pool3_bn (BatchNormalization)  (None, 28, 28, 512)  2048        ['conv3_block12_conc
at[0][0]']
```

```
 pool3_relu (Activation)        (None, 28, 28, 512)  0           ['pool3_bn[0][0]']

 pool3_conv (Conv2D)            (None, 28, 28, 256)  131072      ['pool3_relu[0][0]']

 pool3_pool (AveragePooling2D)  (None, 14, 14, 256)  0           ['pool3_conv[0][0]']

 conv4_block1_0_bn (BatchNormal  (None, 14, 14, 256)  1024       ['pool3_pool[0][0]']
 ization)

 conv4_block1_0_relu (Activatio  (None, 14, 14, 256)  0          ['conv4_block1_0_bn
[0][0]']
 n)

 conv4_block1_1_conv (Conv2D)   (None, 14, 14, 128)  32768       ['conv4_block1_0_rel
u[0][0]']

 conv4_block1_1_bn (BatchNormal  (None, 14, 14, 128)  512        ['conv4_block1_1_con
v[0][0]']
 ization)

 conv4_block1_1_relu (Activatio  (None, 14, 14, 128)  0          ['conv4_block1_1_bn
[0][0]']
 n)

 conv4_block1_2_conv (Conv2D)   (None, 14, 14, 32)   36864       ['conv4_block1_1_rel
u[0][0]']

 conv4_block1_concat (Concatena  (None, 14, 14, 288)  0          ['pool3_pool[0][0]',
 te)                                                             'conv4_block1_2_con
v[0][0]']

 conv4_block2_0_bn (BatchNormal  (None, 14, 14, 288)  1152       ['conv4_block1_conca
t[0][0]']
 ization)

 conv4_block2_0_relu (Activatio  (None, 14, 14, 288)  0          ['conv4_block2_0_bn
[0][0]']
 n)

 conv4_block2_1_conv (Conv2D)   (None, 14, 14, 128)  36864       ['conv4_block2_0_rel
u[0][0]']

 conv4_block2_1_bn (BatchNormal  (None, 14, 14, 128)  512        ['conv4_block2_1_con
v[0][0]']
 ization)

 conv4_block2_1_relu (Activatio  (None, 14, 14, 128)  0          ['conv4_block2_1_bn
[0][0]']
 n)

 conv4_block2_2_conv (Conv2D)   (None, 14, 14, 32)   36864       ['conv4_block2_1_rel
u[0][0]']

 conv4_block2_concat (Concatena  (None, 14, 14, 320)  0          ['conv4_block1_conca
t[0][0]',
 te)                                                             'conv4_block2_2_con
v[0][0]']

 conv4_block3_0_bn (BatchNormal  (None, 14, 14, 320)  1280       ['conv4_block2_conca
t[0][0]']
 ization)

 conv4_block3_0_relu (Activatio  (None, 14, 14, 320)  0          ['conv4_block3_0_bn
[0][0]']
```

```
 n)

 conv4_block3_1_conv (Conv2D)   (None, 14, 14, 128)  40960       ['conv4_block3_0_rel
u[0][0]']

 conv4_block3_1_bn (BatchNormal  (None, 14, 14, 128)  512        ['conv4_block3_1_con
v[0][0]']
 ization)

 conv4_block3_1_relu (Activatio  (None, 14, 14, 128)  0          ['conv4_block3_1_bn
[0][0]']
 n)

 conv4_block3_2_conv (Conv2D)   (None, 14, 14, 32)   36864       ['conv4_block3_1_rel
u[0][0]']

 conv4_block3_concat (Concatena  (None, 14, 14, 352)  0          ['conv4_block2_conca
t[0][0]',
 te)                                                             'conv4_block3_2_con
v[0][0]']

 conv4_block4_0_bn (BatchNormal  (None, 14, 14, 352)  1408       ['conv4_block3_conca
t[0][0]']
 ization)

 conv4_block4_0_relu (Activatio  (None, 14, 14, 352)  0          ['conv4_block4_0_bn
[0][0]']
 n)

 conv4_block4_1_conv (Conv2D)   (None, 14, 14, 128)  45056       ['conv4_block4_0_rel
u[0][0]']

 conv4_block4_1_bn (BatchNormal  (None, 14, 14, 128)  512        ['conv4_block4_1_con
v[0][0]']
 ization)

 conv4_block4_1_relu (Activatio  (None, 14, 14, 128)  0          ['conv4_block4_1_bn
[0][0]']
 n)

 conv4_block4_2_conv (Conv2D)   (None, 14, 14, 32)   36864       ['conv4_block4_1_rel
u[0][0]']

 conv4_block4_concat (Concatena  (None, 14, 14, 384)  0          ['conv4_block3_conca
t[0][0]',
 te)                                                             'conv4_block4_2_con
v[0][0]']

 conv4_block5_0_bn (BatchNormal  (None, 14, 14, 384)  1536       ['conv4_block4_conca
t[0][0]']
 ization)

 conv4_block5_0_relu (Activatio  (None, 14, 14, 384)  0          ['conv4_block5_0_bn
[0][0]']
 n)

 conv4_block5_1_conv (Conv2D)   (None, 14, 14, 128)  49152       ['conv4_block5_0_rel
u[0][0]']

 conv4_block5_1_bn (BatchNormal  (None, 14, 14, 128)  512        ['conv4_block5_1_con
v[0][0]']
 ization)

 conv4_block5_1_relu (Activatio  (None, 14, 14, 128)  0          ['conv4_block5_1_bn
[0][0]']
```

```
 n)

 conv4_block5_2_conv (Conv2D)   (None, 14, 14, 32)   36864       ['conv4_block5_1_rel
u[0][0]']

 conv4_block5_concat (Concatena  (None, 14, 14, 416)  0          ['conv4_block4_conca
t[0][0]',
 te)                                                             'conv4_block5_2_con
v[0][0]']

 conv4_block6_0_bn (BatchNormal  (None, 14, 14, 416)  1664       ['conv4_block5_conca
t[0][0]']
 ization)

 conv4_block6_0_relu (Activatio  (None, 14, 14, 416)  0          ['conv4_block6_0_bn
[0][0]']
 n)

 conv4_block6_1_conv (Conv2D)   (None, 14, 14, 128)  53248       ['conv4_block6_0_rel
u[0][0]']

 conv4_block6_1_bn (BatchNormal  (None, 14, 14, 128)  512        ['conv4_block6_1_con
v[0][0]']
 ization)

 conv4_block6_1_relu (Activatio  (None, 14, 14, 128)  0          ['conv4_block6_1_bn
[0][0]']
 n)

 conv4_block6_2_conv (Conv2D)   (None, 14, 14, 32)   36864       ['conv4_block6_1_rel
u[0][0]']

 conv4_block6_concat (Concatena  (None, 14, 14, 448)  0          ['conv4_block5_conca
t[0][0]',
 te)                                                             'conv4_block6_2_con
v[0][0]']

 conv4_block7_0_bn (BatchNormal  (None, 14, 14, 448)  1792       ['conv4_block6_conca
t[0][0]']
 ization)

 conv4_block7_0_relu (Activatio  (None, 14, 14, 448)  0          ['conv4_block7_0_bn
[0][0]']
 n)

 conv4_block7_1_conv (Conv2D)   (None, 14, 14, 128)  57344       ['conv4_block7_0_rel
u[0][0]']

 conv4_block7_1_bn (BatchNormal  (None, 14, 14, 128)  512        ['conv4_block7_1_con
v[0][0]']
 ization)

 conv4_block7_1_relu (Activatio  (None, 14, 14, 128)  0          ['conv4_block7_1_bn
[0][0]']
 n)

 conv4_block7_2_conv (Conv2D)   (None, 14, 14, 32)   36864       ['conv4_block7_1_rel
u[0][0]']

 conv4_block7_concat (Concatena  (None, 14, 14, 480)  0          ['conv4_block6_conca
t[0][0]',
 te)                                                             'conv4_block7_2_con
v[0][0]']

 conv4_block8_0_bn (BatchNormal  (None, 14, 14, 480)  1920       ['conv4_block7_conca
```

```
t[0][0']
 ization)

 conv4_block8_0_relu (Activatio  (None, 14, 14, 480)  0          ['conv4_block8_0_bn
[0][0']
 n)

 conv4_block8_1_conv (Conv2D)   (None, 14, 14, 128)  61440       ['conv4_block8_0_rel
u[0][0']

 conv4_block8_1_bn (BatchNormal  (None, 14, 14, 128)  512        ['conv4_block8_1_con
v[0][0']
 ization)

 conv4_block8_1_relu (Activatio  (None, 14, 14, 128)  0          ['conv4_block8_1_bn
[0][0']
 n)

 conv4_block8_2_conv (Conv2D)   (None, 14, 14, 32)   36864       ['conv4_block8_1_rel
u[0][0']

 conv4_block8_concat (Concatena  (None, 14, 14, 512)  0          ['conv4_block7_conca
t[0][0',
 te)                                                             'conv4_block8_2_con
v[0][0']

 conv4_block9_0_bn (BatchNormal  (None, 14, 14, 512)  2048       ['conv4_block8_conca
t[0][0']
 ization)

 conv4_block9_0_relu (Activatio  (None, 14, 14, 512)  0          ['conv4_block9_0_bn
[0][0']
 n)

 conv4_block9_1_conv (Conv2D)   (None, 14, 14, 128)  65536       ['conv4_block9_0_rel
u[0][0']

 conv4_block9_1_bn (BatchNormal  (None, 14, 14, 128)  512        ['conv4_block9_1_con
v[0][0']
 ization)

 conv4_block9_1_relu (Activatio  (None, 14, 14, 128)  0          ['conv4_block9_1_bn
[0][0']
 n)

 conv4_block9_2_conv (Conv2D)   (None, 14, 14, 32)   36864       ['conv4_block9_1_rel
u[0][0']

 conv4_block9_concat (Concatena  (None, 14, 14, 544)  0          ['conv4_block8_conca
t[0][0',
 te)                                                             'conv4_block9_2_con
v[0][0']

 conv4_block10_0_bn (BatchNorma  (None, 14, 14, 544)  2176       ['conv4_block9_conca
t[0][0']
 lization)

 conv4_block10_0_relu (Activati  (None, 14, 14, 544)  0          ['conv4_block10_0_bn
[0][0']
 on)

 conv4_block10_1_conv (Conv2D)  (None, 14, 14, 128)  69632       ['conv4_block10_0_re
lu[0][0']

 conv4_block10_1_bn (BatchNorma  (None, 14, 14, 128)  512        ['conv4_block10_1_co
```

```
nv[0][0]']
 lization)

 conv4_block10_1_relu (Activati  (None, 14, 14, 128)  0          ['conv4_block10_1_bn
[0][0]']
 on)

 conv4_block10_2_conv (Conv2D)  (None, 14, 14, 32)   36864       ['conv4_block10_1_re
lu[0][0]']

 conv4_block10_concat (Concaten  (None, 14, 14, 576)  0          ['conv4_block9_conca
t[0][0]',
 ate)                                                 'conv4_block10_2_co
nv[0][0]']

 conv4_block11_0_bn (BatchNorma  (None, 14, 14, 576)  2304       ['conv4_block10_conc
at[0][0]']
 lization)

 conv4_block11_0_relu (Activati  (None, 14, 14, 576)  0          ['conv4_block11_0_bn
[0][0]']
 on)

 conv4_block11_1_conv (Conv2D)  (None, 14, 14, 128)  73728       ['conv4_block11_0_re
lu[0][0]']

 conv4_block11_1_bn (BatchNorma  (None, 14, 14, 128)  512        ['conv4_block11_1_co
nv[0][0]']
 lization)

 conv4_block11_1_relu (Activati  (None, 14, 14, 128)  0          ['conv4_block11_1_bn
[0][0]']
 on)

 conv4_block11_2_conv (Conv2D)  (None, 14, 14, 32)   36864       ['conv4_block11_1_re
lu[0][0]']

 conv4_block11_concat (Concaten  (None, 14, 14, 608)  0          ['conv4_block10_conc
at[0][0]',
 ate)                                                 'conv4_block11_2_co
nv[0][0]']

 conv4_block12_0_bn (BatchNorma  (None, 14, 14, 608)  2432       ['conv4_block11_conc
at[0][0]']
 lization)

 conv4_block12_0_relu (Activati  (None, 14, 14, 608)  0          ['conv4_block12_0_bn
[0][0]']
 on)

 conv4_block12_1_conv (Conv2D)  (None, 14, 14, 128)  77824       ['conv4_block12_0_re
lu[0][0]']

 conv4_block12_1_bn (BatchNorma  (None, 14, 14, 128)  512        ['conv4_block12_1_co
nv[0][0]']
 lization)

 conv4_block12_1_relu (Activati  (None, 14, 14, 128)  0          ['conv4_block12_1_bn
[0][0]']
 on)

 conv4_block12_2_conv (Conv2D)  (None, 14, 14, 32)   36864       ['conv4_block12_1_re
lu[0][0]']

 conv4_block12_concat (Concaten  (None, 14, 14, 640)  0          ['conv4_block11_conc
```

```
at[0][0]',
 ate)                                                              'conv4_block12_2_co
nv[0][0]']

 conv4_block13_0_bn (BatchNorma  (None, 14, 14, 640)  2560       ['conv4_block12_conc
at[0][0]']
 lization)

 conv4_block13_0_relu (Activati  (None, 14, 14, 640)  0          ['conv4_block13_0_bn
[0][0]']
 on)

 conv4_block13_1_conv (Conv2D)  (None, 14, 14, 128)  81920       ['conv4_block13_0_re
lu[0][0]']

 conv4_block13_1_bn (BatchNorma  (None, 14, 14, 128)  512        ['conv4_block13_1_co
nv[0][0]']
 lization)

 conv4_block13_1_relu (Activati  (None, 14, 14, 128)  0          ['conv4_block13_1_bn
[0][0]']
 on)

 conv4_block13_2_conv (Conv2D)  (None, 14, 14, 32)   36864       ['conv4_block13_1_re
lu[0][0]']

 conv4_block13_concat (Concaten  (None, 14, 14, 672)  0          ['conv4_block12_conc
at[0][0]',
 ate)                                                              'conv4_block13_2_co
nv[0][0]']

 conv4_block14_0_bn (BatchNorma  (None, 14, 14, 672)  2688       ['conv4_block13_conc
at[0][0]']
 lization)

 conv4_block14_0_relu (Activati  (None, 14, 14, 672)  0          ['conv4_block14_0_bn
[0][0]']
 on)

 conv4_block14_1_conv (Conv2D)  (None, 14, 14, 128)  86016       ['conv4_block14_0_re
lu[0][0]']

 conv4_block14_1_bn (BatchNorma  (None, 14, 14, 128)  512        ['conv4_block14_1_co
nv[0][0]']
 lization)

 conv4_block14_1_relu (Activati  (None, 14, 14, 128)  0          ['conv4_block14_1_bn
[0][0]']
 on)

 conv4_block14_2_conv (Conv2D)  (None, 14, 14, 32)   36864       ['conv4_block14_1_re
lu[0][0]']

 conv4_block14_concat (Concaten  (None, 14, 14, 704)  0          ['conv4_block13_conc
at[0][0]',
 ate)                                                              'conv4_block14_2_co
nv[0][0]']

 conv4_block15_0_bn (BatchNorma  (None, 14, 14, 704)  2816       ['conv4_block14_conc
at[0][0]']
 lization)

 conv4_block15_0_relu (Activati  (None, 14, 14, 704)  0          ['conv4_block15_0_bn
[0][0]']
 on)
```

```
 conv4_block15_1_conv (Conv2D)  (None, 14, 14, 128)  90112       ['conv4_block15_0_re
lu[0][0]']

 conv4_block15_1_bn (BatchNorma  (None, 14, 14, 128)  512        ['conv4_block15_1_co
nv[0][0]']
 lization)

 conv4_block15_1_relu (Activati  (None, 14, 14, 128)  0          ['conv4_block15_1_bn
[0][0]']
 on)

 conv4_block15_2_conv (Conv2D)  (None, 14, 14, 32)   36864       ['conv4_block15_1_re
lu[0][0]']

 conv4_block15_concat (Concaten  (None, 14, 14, 736)  0          ['conv4_block14_conc
at[0][0]',
 ate)                                                            'conv4_block15_2_co
nv[0][0]']

 conv4_block16_0_bn (BatchNorma  (None, 14, 14, 736)  2944       ['conv4_block15_conc
at[0][0]']
 lization)

 conv4_block16_0_relu (Activati  (None, 14, 14, 736)  0          ['conv4_block16_0_bn
[0][0]']
 on)

 conv4_block16_1_conv (Conv2D)  (None, 14, 14, 128)  94208       ['conv4_block16_0_re
lu[0][0]']

 conv4_block16_1_bn (BatchNorma  (None, 14, 14, 128)  512        ['conv4_block16_1_co
nv[0][0]']
 lization)

 conv4_block16_1_relu (Activati  (None, 14, 14, 128)  0          ['conv4_block16_1_bn
[0][0]']
 on)

 conv4_block16_2_conv (Conv2D)  (None, 14, 14, 32)   36864       ['conv4_block16_1_re
lu[0][0]']

 conv4_block16_concat (Concaten  (None, 14, 14, 768)  0          ['conv4_block15_conc
at[0][0]',
 ate)                                                            'conv4_block16_2_co
nv[0][0]']

 conv4_block17_0_bn (BatchNorma  (None, 14, 14, 768)  3072       ['conv4_block16_conc
at[0][0]']
 lization)

 conv4_block17_0_relu (Activati  (None, 14, 14, 768)  0          ['conv4_block17_0_bn
[0][0]']
 on)

 conv4_block17_1_conv (Conv2D)  (None, 14, 14, 128)  98304       ['conv4_block17_0_re
lu[0][0]']

 conv4_block17_1_bn (BatchNorma  (None, 14, 14, 128)  512        ['conv4_block17_1_co
nv[0][0]']
 lization)

 conv4_block17_1_relu (Activati  (None, 14, 14, 128)  0          ['conv4_block17_1_bn
[0][0]']
 on)
```

```
 conv4_block17_2_conv (Conv2D)  (None, 14, 14, 32)   36864       ['conv4_block17_1_re
lu[0][0]']

 conv4_block17_concat (Concaten  (None, 14, 14, 800)  0          ['conv4_block16_conc
at[0][0]',
 ate)                                                             'conv4_block17_2_co
nv[0][0]']

 conv4_block18_0_bn (BatchNorma  (None, 14, 14, 800)  3200       ['conv4_block17_conc
at[0][0]']
 lization)

 conv4_block18_0_relu (Activati  (None, 14, 14, 800)  0          ['conv4_block18_0_bn
[0][0]']
 on)

 conv4_block18_1_conv (Conv2D)  (None, 14, 14, 128)  102400      ['conv4_block18_0_re
lu[0][0]']

 conv4_block18_1_bn (BatchNorma  (None, 14, 14, 128)  512        ['conv4_block18_1_co
nv[0][0]']
 lization)

 conv4_block18_1_relu (Activati  (None, 14, 14, 128)  0          ['conv4_block18_1_bn
[0][0]']
 on)

 conv4_block18_2_conv (Conv2D)  (None, 14, 14, 32)   36864       ['conv4_block18_1_re
lu[0][0]']

 conv4_block18_concat (Concaten  (None, 14, 14, 832)  0          ['conv4_block17_conc
at[0][0]',
 ate)                                                             'conv4_block18_2_co
nv[0][0]']

 conv4_block19_0_bn (BatchNorma  (None, 14, 14, 832)  3328       ['conv4_block18_conc
at[0][0]']
 lization)

 conv4_block19_0_relu (Activati  (None, 14, 14, 832)  0          ['conv4_block19_0_bn
[0][0]']
 on)

 conv4_block19_1_conv (Conv2D)  (None, 14, 14, 128)  106496      ['conv4_block19_0_re
lu[0][0]']

 conv4_block19_1_bn (BatchNorma  (None, 14, 14, 128)  512        ['conv4_block19_1_co
nv[0][0]']
 lization)

 conv4_block19_1_relu (Activati  (None, 14, 14, 128)  0          ['conv4_block19_1_bn
[0][0]']
 on)

 conv4_block19_2_conv (Conv2D)  (None, 14, 14, 32)   36864       ['conv4_block19_1_re
lu[0][0]']

 conv4_block19_concat (Concaten  (None, 14, 14, 864)  0          ['conv4_block18_conc
at[0][0]',
 ate)                                                             'conv4_block19_2_co
nv[0][0]']

 conv4_block20_0_bn (BatchNorma  (None, 14, 14, 864)  3456       ['conv4_block19_conc
at[0][0]']
```

```
 lization)

 conv4_block20_0_relu (Activati  (None, 14, 14, 864)  0          ['conv4_block20_0_bn
[0][0]']
 on)

 conv4_block20_1_conv (Conv2D)  (None, 14, 14, 128)  110592      ['conv4_block20_0_re
lu[0][0]']

 conv4_block20_1_bn (BatchNorma  (None, 14, 14, 128)  512        ['conv4_block20_1_co
nv[0][0]']
 lization)

 conv4_block20_1_relu (Activati  (None, 14, 14, 128)  0          ['conv4_block20_1_bn
[0][0]']
 on)

 conv4_block20_2_conv (Conv2D)  (None, 14, 14, 32)   36864       ['conv4_block20_1_re
lu[0][0]']

 conv4_block20_concat (Concaten  (None, 14, 14, 896)  0          ['conv4_block19_conc
at[0][0]',
 ate)                                                            'conv4_block20_2_co
nv[0][0]']

 conv4_block21_0_bn (BatchNorma  (None, 14, 14, 896)  3584       ['conv4_block20_conc
at[0][0]']
 lization)

 conv4_block21_0_relu (Activati  (None, 14, 14, 896)  0          ['conv4_block21_0_bn
[0][0]']
 on)

 conv4_block21_1_conv (Conv2D)  (None, 14, 14, 128)  114688      ['conv4_block21_0_re
lu[0][0]']

 conv4_block21_1_bn (BatchNorma  (None, 14, 14, 128)  512        ['conv4_block21_1_co
nv[0][0]']
 lization)

 conv4_block21_1_relu (Activati  (None, 14, 14, 128)  0          ['conv4_block21_1_bn
[0][0]']
 on)

 conv4_block21_2_conv (Conv2D)  (None, 14, 14, 32)   36864       ['conv4_block21_1_re
lu[0][0]']

 conv4_block21_concat (Concaten  (None, 14, 14, 928)  0          ['conv4_block20_conc
at[0][0]',
 ate)                                                            'conv4_block21_2_co
nv[0][0]']

 conv4_block22_0_bn (BatchNorma  (None, 14, 14, 928)  3712       ['conv4_block21_conc
at[0][0]']
 lization)

 conv4_block22_0_relu (Activati  (None, 14, 14, 928)  0          ['conv4_block22_0_bn
[0][0]']
 on)

 conv4_block22_1_conv (Conv2D)  (None, 14, 14, 128)  118784      ['conv4_block22_0_re
lu[0][0]']

 conv4_block22_1_bn (BatchNorma  (None, 14, 14, 128)  512        ['conv4_block22_1_co
nv[0][0]']
```

```
 lization)

 conv4_block22_1_relu (Activati  (None, 14, 14, 128)  0          ['conv4_block22_1_bn
[0][0]']
 on)

 conv4_block22_2_conv (Conv2D)  (None, 14, 14, 32)   36864       ['conv4_block22_1_re
lu[0][0]']

 conv4_block22_concat (Concaten  (None, 14, 14, 960)  0          ['conv4_block21_conc
at[0][0]',
 ate)                                                             'conv4_block22_2_co
nv[0][0]']

 conv4_block23_0_bn (BatchNorma  (None, 14, 14, 960)  3840       ['conv4_block22_conc
at[0][0]']
 lization)

 conv4_block23_0_relu (Activati  (None, 14, 14, 960)  0          ['conv4_block23_0_bn
[0][0]']
 on)

 conv4_block23_1_conv (Conv2D)  (None, 14, 14, 128)  122880      ['conv4_block23_0_re
lu[0][0]']

 conv4_block23_1_bn (BatchNorma  (None, 14, 14, 128)  512        ['conv4_block23_1_co
nv[0][0]']
 lization)

 conv4_block23_1_relu (Activati  (None, 14, 14, 128)  0          ['conv4_block23_1_bn
[0][0]']
 on)

 conv4_block23_2_conv (Conv2D)  (None, 14, 14, 32)   36864       ['conv4_block23_1_re
lu[0][0]']

 conv4_block23_concat (Concaten  (None, 14, 14, 992)  0          ['conv4_block22_conc
at[0][0]',
 ate)                                                             'conv4_block23_2_co
nv[0][0]']

 conv4_block24_0_bn (BatchNorma  (None, 14, 14, 992)  3968       ['conv4_block23_conc
at[0][0]']
 lization)

 conv4_block24_0_relu (Activati  (None, 14, 14, 992)  0          ['conv4_block24_0_bn
[0][0]']
 on)

 conv4_block24_1_conv (Conv2D)  (None, 14, 14, 128)  126976      ['conv4_block24_0_re
lu[0][0]']

 conv4_block24_1_bn (BatchNorma  (None, 14, 14, 128)  512        ['conv4_block24_1_co
nv[0][0]']
 lization)

 conv4_block24_1_relu (Activati  (None, 14, 14, 128)  0          ['conv4_block24_1_bn
[0][0]']
 on)

 conv4_block24_2_conv (Conv2D)  (None, 14, 14, 32)   36864       ['conv4_block24_1_re
lu[0][0]']

 conv4_block24_concat (Concaten  (None, 14, 14, 1024  0          ['conv4_block23_conc
at[0][0]',
```

```
 ate)                           )                                  'conv4_block24_2_co
nv[0][0]']

 pool4_bn (BatchNormalization)  (None, 14, 14, 1024  4096        ['conv4_block24_conc
at[0][0]']
                                )

 pool4_relu (Activation)        (None, 14, 14, 1024  0           ['pool4_bn[0][0]']
                                )

 pool4_conv (Conv2D)            (None, 14, 14, 512)  524288      ['pool4_relu[0][0]']

 pool4_pool (AveragePooling2D)  (None, 7, 7, 512)    0           ['pool4_conv[0][0]']

 conv5_block1_0_bn (BatchNormal  (None, 7, 7, 512)   2048        ['pool4_pool[0][0]']
 ization)

 conv5_block1_0_relu (Activatio  (None, 7, 7, 512)   0           ['conv5_block1_0_bn
[0][0]']
 n)

 conv5_block1_1_conv (Conv2D)   (None, 7, 7, 128)    65536       ['conv5_block1_0_rel
u[0][0]']

 conv5_block1_1_bn (BatchNormal  (None, 7, 7, 128)   512         ['conv5_block1_1_con
v[0][0]']
 ization)

 conv5_block1_1_relu (Activatio  (None, 7, 7, 128)   0           ['conv5_block1_1_bn
[0][0]']
 n)

 conv5_block1_2_conv (Conv2D)   (None, 7, 7, 32)     36864       ['conv5_block1_1_rel
u[0][0]']

 conv5_block1_concat (Concatena  (None, 7, 7, 544)   0           ['pool4_pool[0][0]',
 te)                                                             'conv5_block1_2_con
v[0][0]']

 conv5_block2_0_bn (BatchNormal  (None, 7, 7, 544)   2176        ['conv5_block1_conca
t[0][0]']
 ization)

 conv5_block2_0_relu (Activatio  (None, 7, 7, 544)   0           ['conv5_block2_0_bn
[0][0]']
 n)

 conv5_block2_1_conv (Conv2D)   (None, 7, 7, 128)    69632       ['conv5_block2_0_rel
u[0][0]']

 conv5_block2_1_bn (BatchNormal  (None, 7, 7, 128)   512         ['conv5_block2_1_con
v[0][0]']
 ization)

 conv5_block2_1_relu (Activatio  (None, 7, 7, 128)   0           ['conv5_block2_1_bn
[0][0]']
 n)

 conv5_block2_2_conv (Conv2D)   (None, 7, 7, 32)     36864       ['conv5_block2_1_rel
u[0][0]']

 conv5_block2_concat (Concatena  (None, 7, 7, 576)   0           ['conv5_block1_conca
t[0][0]',
 te)                                                             'conv5_block2_2_con
v[0][0]']
```

```
 conv5_block3_0_bn (BatchNormal  (None, 7, 7, 576)   2304        ['conv5_block2_conca
t[0][0]']
 ization)

 conv5_block3_0_relu (Activatio  (None, 7, 7, 576)   0           ['conv5_block3_0_bn
[0][0]']
 n)

 conv5_block3_1_conv (Conv2D)   (None, 7, 7, 128)    73728       ['conv5_block3_0_rel
u[0][0]']

 conv5_block3_1_bn (BatchNormal  (None, 7, 7, 128)   512         ['conv5_block3_1_con
v[0][0]']
 ization)

 conv5_block3_1_relu (Activatio  (None, 7, 7, 128)   0           ['conv5_block3_1_bn
[0][0]']
 n)

 conv5_block3_2_conv (Conv2D)   (None, 7, 7, 32)     36864       ['conv5_block3_1_rel
u[0][0]']

 conv5_block3_concat (Concatena  (None, 7, 7, 608)   0           ['conv5_block2_conca
t[0][0]',
 te)                                                             'conv5_block3_2_con
v[0][0]']

 conv5_block4_0_bn (BatchNormal  (None, 7, 7, 608)   2432        ['conv5_block3_conca
t[0][0]']
 ization)

 conv5_block4_0_relu (Activatio  (None, 7, 7, 608)   0           ['conv5_block4_0_bn
[0][0]']
 n)

 conv5_block4_1_conv (Conv2D)   (None, 7, 7, 128)    77824       ['conv5_block4_0_rel
u[0][0]']

 conv5_block4_1_bn (BatchNormal  (None, 7, 7, 128)   512         ['conv5_block4_1_con
v[0][0]']
 ization)

 conv5_block4_1_relu (Activatio  (None, 7, 7, 128)   0           ['conv5_block4_1_bn
[0][0]']
 n)

 conv5_block4_2_conv (Conv2D)   (None, 7, 7, 32)     36864       ['conv5_block4_1_rel
u[0][0]']

 conv5_block4_concat (Concatena  (None, 7, 7, 640)   0           ['conv5_block3_conca
t[0][0]',
 te)                                                             'conv5_block4_2_con
v[0][0]']

 conv5_block5_0_bn (BatchNormal  (None, 7, 7, 640)   2560        ['conv5_block4_conca
t[0][0]']
 ization)

 conv5_block5_0_relu (Activatio  (None, 7, 7, 640)   0           ['conv5_block5_0_bn
[0][0]']
 n)

 conv5_block5_1_conv (Conv2D)   (None, 7, 7, 128)    81920       ['conv5_block5_0_rel
u[0][0]']
```

```
 conv5_block5_1_bn (BatchNormal  (None, 7, 7, 128)   512         ['conv5_block5_1_con
v[0][0]']
 ization)

 conv5_block5_1_relu (Activatio  (None, 7, 7, 128)   0           ['conv5_block5_1_bn
[0][0]']
 n)

 conv5_block5_2_conv (Conv2D)   (None, 7, 7, 32)     36864       ['conv5_block5_1_rel
u[0][0]']

 conv5_block5_concat (Concatena  (None, 7, 7, 672)   0           ['conv5_block4_conca
t[0][0]',
 te)                                                 'conv5_block5_2_con
v[0][0]']

 conv5_block6_0_bn (BatchNormal  (None, 7, 7, 672)   2688        ['conv5_block5_conca
t[0][0]']
 ization)

 conv5_block6_0_relu (Activatio  (None, 7, 7, 672)   0           ['conv5_block6_0_bn
[0][0]']
 n)

 conv5_block6_1_conv (Conv2D)   (None, 7, 7, 128)    86016       ['conv5_block6_0_rel
u[0][0]']

 conv5_block6_1_bn (BatchNormal  (None, 7, 7, 128)   512         ['conv5_block6_1_con
v[0][0]']
 ization)

 conv5_block6_1_relu (Activatio  (None, 7, 7, 128)   0           ['conv5_block6_1_bn
[0][0]']
 n)

 conv5_block6_2_conv (Conv2D)   (None, 7, 7, 32)     36864       ['conv5_block6_1_rel
u[0][0]']

 conv5_block6_concat (Concatena  (None, 7, 7, 704)   0           ['conv5_block5_conca
t[0][0]',
 te)                                                 'conv5_block6_2_con
v[0][0]']

 conv5_block7_0_bn (BatchNormal  (None, 7, 7, 704)   2816        ['conv5_block6_conca
t[0][0]']
 ization)

 conv5_block7_0_relu (Activatio  (None, 7, 7, 704)   0           ['conv5_block7_0_bn
[0][0]']
 n)

 conv5_block7_1_conv (Conv2D)   (None, 7, 7, 128)    90112       ['conv5_block7_0_rel
u[0][0]']

 conv5_block7_1_bn (BatchNormal  (None, 7, 7, 128)   512         ['conv5_block7_1_con
v[0][0]']
 ization)

 conv5_block7_1_relu (Activatio  (None, 7, 7, 128)   0           ['conv5_block7_1_bn
[0][0]']
 n)

 conv5_block7_2_conv (Conv2D)   (None, 7, 7, 32)     36864       ['conv5_block7_1_rel
u[0][0]']
```

```
 conv5_block7_concat (Concatena  (None, 7, 7, 736)   0           ['conv5_block6_concat[0][0]',
 te)                                                              'conv5_block7_2_conv[0][0]']

 conv5_block8_0_bn (BatchNormal  (None, 7, 7, 736)   2944        ['conv5_block7_concat[0][0]']
 ization)

 conv5_block8_0_relu (Activatio  (None, 7, 7, 736)   0           ['conv5_block8_0_bn[0][0]']
 n)

 conv5_block8_1_conv (Conv2D)   (None, 7, 7, 128)    94208       ['conv5_block8_0_relu[0][0]']

 conv5_block8_1_bn (BatchNormal  (None, 7, 7, 128)   512         ['conv5_block8_1_conv[0][0]']
 ization)

 conv5_block8_1_relu (Activatio  (None, 7, 7, 128)   0           ['conv5_block8_1_bn[0][0]']
 n)

 conv5_block8_2_conv (Conv2D)   (None, 7, 7, 32)     36864       ['conv5_block8_1_relu[0][0]']

 conv5_block8_concat (Concatena  (None, 7, 7, 768)   0           ['conv5_block7_concat[0][0]',
 te)                                                              'conv5_block8_2_conv[0][0]']

 conv5_block9_0_bn (BatchNormal  (None, 7, 7, 768)   3072        ['conv5_block8_concat[0][0]']
 ization)

 conv5_block9_0_relu (Activatio  (None, 7, 7, 768)   0           ['conv5_block9_0_bn[0][0]']
 n)

 conv5_block9_1_conv (Conv2D)   (None, 7, 7, 128)    98304       ['conv5_block9_0_relu[0][0]']

 conv5_block9_1_bn (BatchNormal  (None, 7, 7, 128)   512         ['conv5_block9_1_conv[0][0]']
 ization)

 conv5_block9_1_relu (Activatio  (None, 7, 7, 128)   0           ['conv5_block9_1_bn[0][0]']
 n)

 conv5_block9_2_conv (Conv2D)   (None, 7, 7, 32)     36864       ['conv5_block9_1_relu[0][0]']

 conv5_block9_concat (Concatena  (None, 7, 7, 800)   0           ['conv5_block8_concat[0][0]',
 te)                                                              'conv5_block9_2_conv[0][0]']

 conv5_block10_0_bn (BatchNorma  (None, 7, 7, 800)   3200        ['conv5_block9_concat[0][0]']
 lization)

 conv5_block10_0_relu (Activati  (None, 7, 7, 800)   0           ['conv5_block10_0_bn
```

```
[0][0]']
 on)

 conv5_block10_1_conv (Conv2D)  (None, 7, 7, 128)    102400      ['conv5_block10_0_re
lu[0][0]']

 conv5_block10_1_bn (BatchNorma  (None, 7, 7, 128)   512         ['conv5_block10_1_co
nv[0][0]']
 lization)

 conv5_block10_1_relu (Activati  (None, 7, 7, 128)   0           ['conv5_block10_1_bn
[0][0]']
 on)

 conv5_block10_2_conv (Conv2D)  (None, 7, 7, 32)     36864       ['conv5_block10_1_re
lu[0][0]']

 conv5_block10_concat (Concaten  (None, 7, 7, 832)   0           ['conv5_block9_conca
t[0][0]',
 ate)                                                 'conv5_block10_2_co
nv[0][0]']

 conv5_block11_0_bn (BatchNorma  (None, 7, 7, 832)   3328        ['conv5_block10_conc
at[0][0]']
 lization)

 conv5_block11_0_relu (Activati  (None, 7, 7, 832)   0           ['conv5_block11_0_bn
[0][0]']
 on)

 conv5_block11_1_conv (Conv2D)  (None, 7, 7, 128)    106496      ['conv5_block11_0_re
lu[0][0]']

 conv5_block11_1_bn (BatchNorma  (None, 7, 7, 128)   512         ['conv5_block11_1_co
nv[0][0]']
 lization)

 conv5_block11_1_relu (Activati  (None, 7, 7, 128)   0           ['conv5_block11_1_bn
[0][0]']
 on)

 conv5_block11_2_conv (Conv2D)  (None, 7, 7, 32)     36864       ['conv5_block11_1_re
lu[0][0]']

 conv5_block11_concat (Concaten  (None, 7, 7, 864)   0           ['conv5_block10_conc
at[0][0]',
 ate)                                                 'conv5_block11_2_co
nv[0][0]']

 conv5_block12_0_bn (BatchNorma  (None, 7, 7, 864)   3456        ['conv5_block11_conc
at[0][0]']
 lization)

 conv5_block12_0_relu (Activati  (None, 7, 7, 864)   0           ['conv5_block12_0_bn
[0][0]']
 on)

 conv5_block12_1_conv (Conv2D)  (None, 7, 7, 128)    110592      ['conv5_block12_0_re
lu[0][0]']

 conv5_block12_1_bn (BatchNorma  (None, 7, 7, 128)   512         ['conv5_block12_1_co
nv[0][0]']
 lization)

 conv5_block12_1_relu (Activati  (None, 7, 7, 128)   0           ['conv5_block12_1_bn
```

```
[0][0]']
 on)

 conv5_block12_2_conv (Conv2D)  (None, 7, 7, 32)     36864       ['conv5_block12_1_re
lu[0][0]']

 conv5_block12_concat (Concaten  (None, 7, 7, 896)   0           ['conv5_block11_conc
at[0][0]',
 ate)                                                 'conv5_block12_2_co
nv[0][0]']

 conv5_block13_0_bn (BatchNorma  (None, 7, 7, 896)   3584        ['conv5_block12_conc
at[0][0]']
 lization)

 conv5_block13_0_relu (Activati  (None, 7, 7, 896)   0           ['conv5_block13_0_bn
[0][0]']
 on)

 conv5_block13_1_conv (Conv2D)  (None, 7, 7, 128)    114688      ['conv5_block13_0_re
lu[0][0]']

 conv5_block13_1_bn (BatchNorma  (None, 7, 7, 128)   512         ['conv5_block13_1_co
nv[0][0]']
 lization)

 conv5_block13_1_relu (Activati  (None, 7, 7, 128)   0           ['conv5_block13_1_bn
[0][0]']
 on)

 conv5_block13_2_conv (Conv2D)  (None, 7, 7, 32)     36864       ['conv5_block13_1_re
lu[0][0]']

 conv5_block13_concat (Concaten  (None, 7, 7, 928)   0           ['conv5_block12_conc
at[0][0]',
 ate)                                                 'conv5_block13_2_co
nv[0][0]']

 conv5_block14_0_bn (BatchNorma  (None, 7, 7, 928)   3712        ['conv5_block13_conc
at[0][0]']
 lization)

 conv5_block14_0_relu (Activati  (None, 7, 7, 928)   0           ['conv5_block14_0_bn
[0][0]']
 on)

 conv5_block14_1_conv (Conv2D)  (None, 7, 7, 128)    118784      ['conv5_block14_0_re
lu[0][0]']

 conv5_block14_1_bn (BatchNorma  (None, 7, 7, 128)   512         ['conv5_block14_1_co
nv[0][0]']
 lization)

 conv5_block14_1_relu (Activati  (None, 7, 7, 128)   0           ['conv5_block14_1_bn
[0][0]']
 on)

 conv5_block14_2_conv (Conv2D)  (None, 7, 7, 32)     36864       ['conv5_block14_1_re
lu[0][0]']

 conv5_block14_concat (Concaten  (None, 7, 7, 960)   0           ['conv5_block13_conc
at[0][0]',
 ate)                                                 'conv5_block14_2_co
nv[0][0]']
```

```
 conv5_block15_0_bn (BatchNorma  (None, 7, 7, 960)   3840        ['conv5_block14_conc
at[0][0]']
 lization)

 conv5_block15_0_relu (Activati  (None, 7, 7, 960)   0           ['conv5_block15_0_bn
[0][0]']
 on)

 conv5_block15_1_conv (Conv2D)  (None, 7, 7, 128)    122880      ['conv5_block15_0_re
lu[0][0]']

 conv5_block15_1_bn (BatchNorma  (None, 7, 7, 128)   512         ['conv5_block15_1_co
nv[0][0]']
 lization)

 conv5_block15_1_relu (Activati  (None, 7, 7, 128)   0           ['conv5_block15_1_bn
[0][0]']
 on)

 conv5_block15_2_conv (Conv2D)  (None, 7, 7, 32)     36864       ['conv5_block15_1_re
lu[0][0]']

 conv5_block15_concat (Concaten  (None, 7, 7, 992)   0           ['conv5_block14_conc
at[0][0]',
 ate)                                                 'conv5_block15_2_co
nv[0][0]']

 conv5_block16_0_bn (BatchNorma  (None, 7, 7, 992)   3968        ['conv5_block15_conc
at[0][0]']
 lization)

 conv5_block16_0_relu (Activati  (None, 7, 7, 992)   0           ['conv5_block16_0_bn
[0][0]']
 on)

 conv5_block16_1_conv (Conv2D)  (None, 7, 7, 128)    126976      ['conv5_block16_0_re
lu[0][0]']

 conv5_block16_1_bn (BatchNorma  (None, 7, 7, 128)   512         ['conv5_block16_1_co
nv[0][0]']
 lization)

 conv5_block16_1_relu (Activati  (None, 7, 7, 128)   0           ['conv5_block16_1_bn
[0][0]']
 on)

 conv5_block16_2_conv (Conv2D)  (None, 7, 7, 32)     36864       ['conv5_block16_1_re
lu[0][0]']

 conv5_block16_concat (Concaten  (None, 7, 7, 1024)  0           ['conv5_block15_conc
at[0][0]',
 ate)                                                 'conv5_block16_2_co
nv[0][0]']

 bn (BatchNormalization)        (None, 7, 7, 1024)   4096        ['conv5_block16_conc
at[0][0]']

 relu (Activation)              (None, 7, 7, 1024)   0           ['bn[0][0]']

 dropout_6 (Dropout)            (None, 7, 7, 1024)   0           ['relu[0][0]']

 flatten_2 (Flatten)            (None, 50176)        0           ['dropout_6[0][0]']

 dense_5 (Dense)                (None, 128)          6422656     ['flatten_2[0][0]']
```

```
 dense_6 (Dense)                (None, 3)            387         ['dense_5[0][0]']

==================================================================================================
Total params: 13,460,547
Trainable params: 6,423,043
Non-trainable params: 7,037,504
__________________________________________________________________________________________________
```

7. Use Early Stopping on the validation loss with a patience of 2 epochs to prevent overfitting

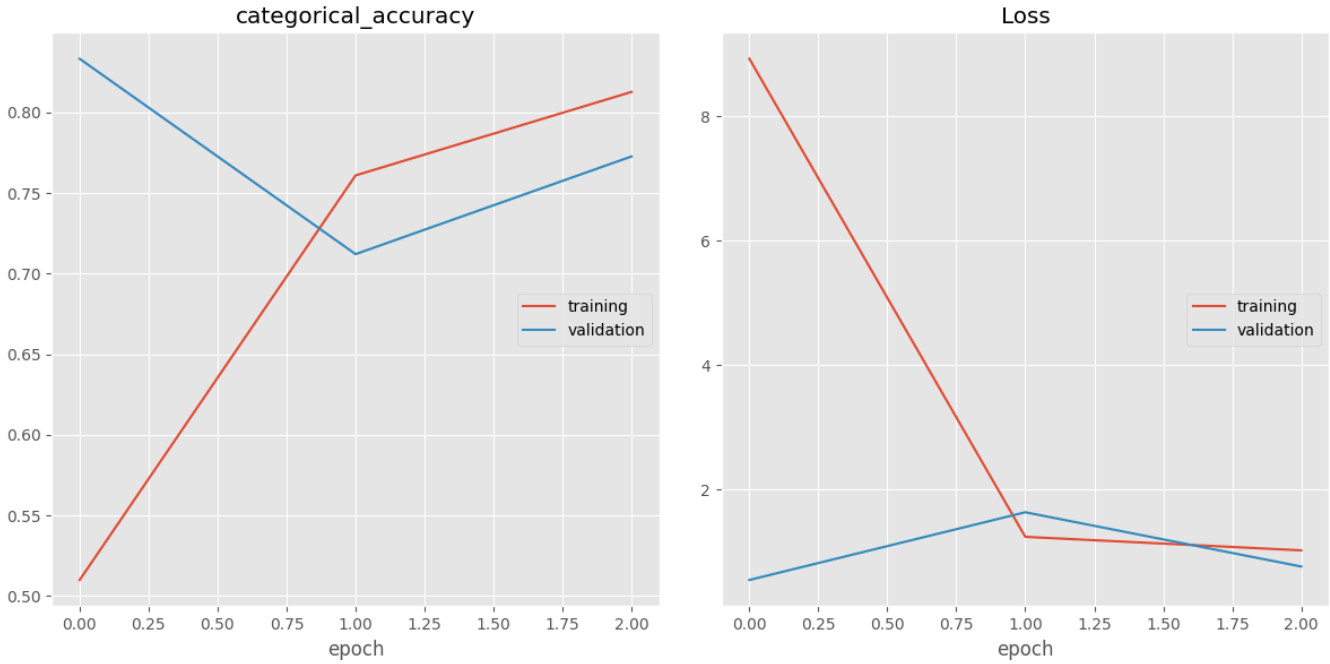
8. Use 15 epochs with a batch size of 7 - tinker with these to optimize accuracy

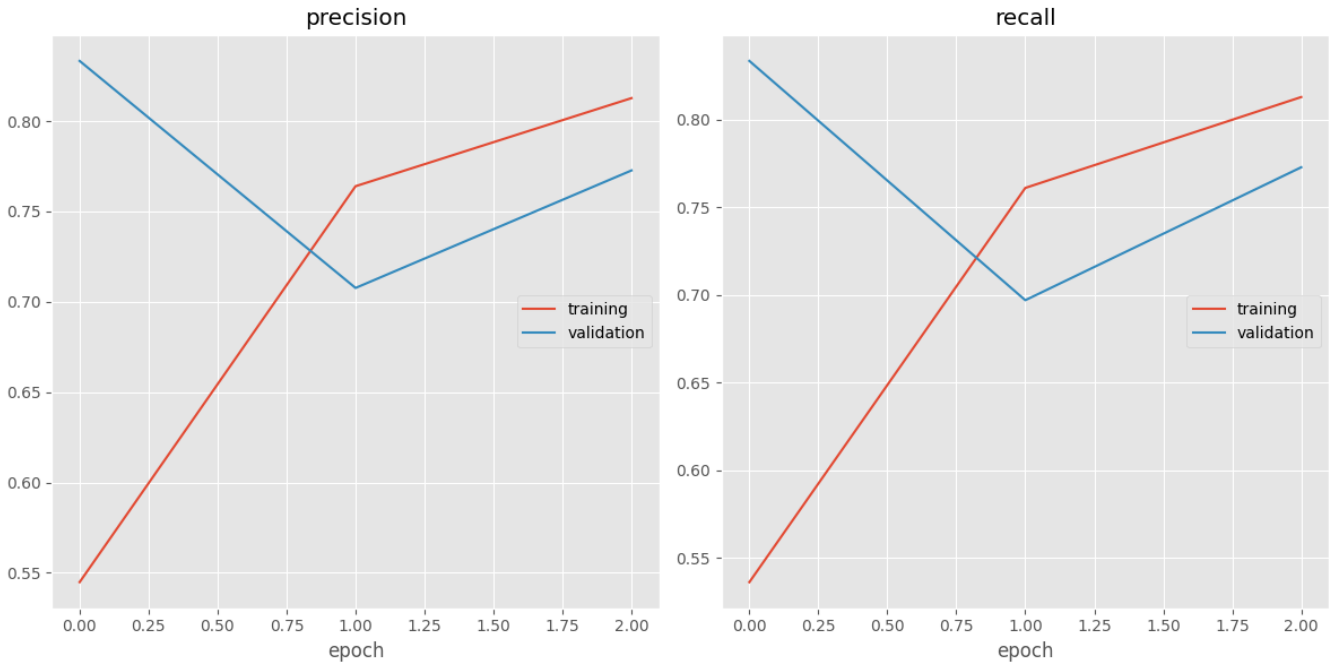
In [32]:
```
es = EarlyStopping(monitor='val_loss', mode='min', verbose=1, patience=2)
mc = ModelCheckpoint('./model_objects/densenet_model.h5', monitor='val_loss', mode='m
EPOCHS = 10
BATCH_SIZE = 7
```

9. Train using an image generator and test the accuracy on the test data at each epoch

10. Plot the training & validation accuracy & loss

In [33]:
```
densenet_model.fit(
    densenet_train_generator,
    epochs=EPOCHS,
    batch_size=BATCH_SIZE,
    validation_data=densenet_test_generator,
    callbacks=[
        es,
        mc,
        PlotLossesKerasTF()
    ]
)
```






```
categorical_accuracy
        training                 (min:    0.510, max:    0.813, cur:    0.813)
        validation               (min:    0.712, max:    0.833, cur:    0.773)
Loss
        training                 (min:    1.026, max:    8.920, cur:    1.026)
        validation               (min:    0.550, max:    1.637, cur:    0.766)
precision
        training                 (min:    0.545, max:    0.813, cur:    0.813)
        validation               (min:    0.708, max:    0.833, cur:    0.773)
recall
        training                 (min:    0.536, max:    0.813, cur:    0.813)
        validation               (min:    0.697, max:    0.833, cur:    0.773)
16/16 [==============================] - 25s 2s/step - loss: 1.0257 - categorical_acc
uracy: 0.8127 - precision: 0.8127 - recall: 0.8127 - val_loss: 0.7657 - val_categoric
al_accuracy: 0.7727 - val_precision: 0.7727 - val_recall: 0.7727
Epoch 3: early stopping
```

Out[33]: `<keras.callbacks.History at 0x13a849780>`

11. Observe metrics Precision, Recall, F1-Score for all classes on both grayscale & color models - determine if the classes are good.

In [34]:
```
# Use sklearn classification report to evaluate the densenet model
best_densenet_model = load_model('./model_objects/densenet_model.h5')
y_pred = best_densenet_model.predict(densenet_test_generator)
y_pred = np.argmax(y_pred, axis=1)
```

```
y_true = densenet_test_generator.classes
print(classification_report(y_true, y_pred, target_names=classes))
print(f'Precision:\t{precision_score(y_true, y_pred, average="macro")}')
print(f'Recall:\t\t{recall_score(y_true, y_pred, average="macro")}')
print(f'F1 Score:\t{f1_score(y_true, y_pred, average="macro")}')

# Definitievely the best model so far despite lower categorical accuracy (0.697 on th
```

```
5/5 [==============================] - 7s 950ms/step
                precision    recall  f1-score   support

       Healthy       0.31      0.45      0.37        20
Type 1 disease       0.23      0.23      0.23        26
Type 2 disease       0.18      0.10      0.13        20

      accuracy                           0.26        66
     macro avg       0.24      0.26      0.24        66
  weighted avg       0.24      0.26      0.24        66

Precision:      0.2409774133912065
Recall:         0.2602564102564103
F1 Score:       0.24238280920308572
```

Section 5: Final Step

1. Compare all of the models on the basis of accuracy, precision, recall, f1-score

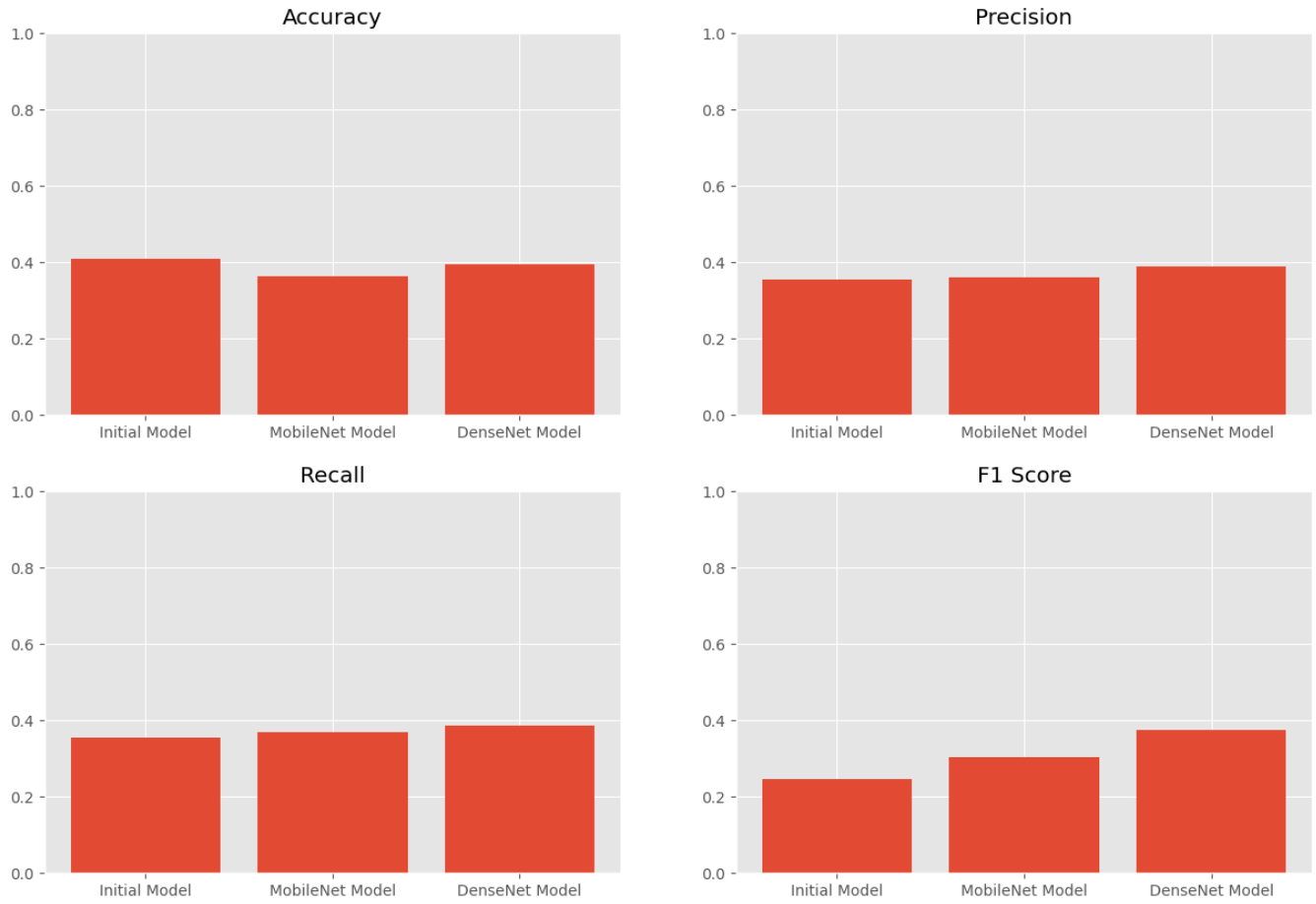
In [42]:
```
# Write a function to evaluate the models and store their metrics in an output dictio
def evaluate_model(model, model_name, test_generator, output_dict):
    y_pred = model.predict(test_generator)
    y_pred = np.argmax(y_pred, axis=1)
    y_true = test_generator.classes
    output_dict['Accuracy'][model_name] = accuracy_score(y_true, y_pred)
    output_dict['Precision'][model_name] = precision_score(y_true, y_pred, average='m
    output_dict['Recall'][model_name] = recall_score(y_true, y_pred, average='macro')
    output_dict['F1 Score'][model_name] = f1_score(y_true, y_pred, average='macro')
```

In [45]:
```
# Create a dictionary to store the key metrics for each model
metric_dict = {
    'Accuracy': {},
    'Precision': {},
    'Recall': {},
    'F1 Score': {}
}

# Loop over the models and evaluate them, storing the metrics in the dictionary
for model, model_name, generator in zip(
    [best_model, best_mobilenet_model, best_densenet_model],
    ['Initial Model', 'MobileNet Model', 'DenseNet Model'],
    [test_generator, mobilenet_test_generator, densenet_test_generator]
):
    evaluate_model(model, model_name, generator, metric_dict)
```

```
5/5 [==============================] - 1s 219ms/step
/usr/local/lib/python3.10/site-packages/sklearn/metrics/_classification.py:1334: Unde
finedMetricWarning: Precision is ill-defined and being set to 0.0 in labels with no p
redicted samples. Use `zero_division` parameter to control this behavior.
  _warn_prf(average, modifier, msg_start, len(result))
5/5 [==============================] - 2s 301ms/step
5/5 [==============================] - 5s 773ms/step
```

In [46]:
```
# Plot each of the metrics for each model
fig, axes = plt.subplots(2, 2, figsize=(15, 10))
for i, metric in enumerate(metric_dict.keys()):
```

```
    ax = axes[i//2, i%2]
    ax.bar(metric_dict[metric].keys(), metric_dict[metric].values())
    ax.set_title(metric)
    ax.set_ylim(0, 1)
plt.show()
```



Model Notes:

- The intial model was the most accurate simply because it predicted the majority class every time
- The DenseNet model was able to achieve slightly better Recall & Precision leading to a much better F1 Score
- All of the models do not perform very well...

Model Improvements:

- I'd probably use some class weights in the modeling steps to further highlight the healthy sets of lungs. Even though the test & train sets both had mostly diseased pictures, I'm not sure that's how it would be in practice
- I'd also optimize even more for F1-Score, particularly in the initial model
- I'd love to grayscale the training images for the transfer learning base models, but alas