Imported Libraries

In [1]:
```
import tensorflow as tf
import keras
from tensorflow.keras.models import Sequential, Model
from tensorflow.keras.layers import Dense, Conv2D , MaxPool2D , Flatten , Dropout, BatchNormalization, LSTM, Input, Re
shape
from tensorflow.keras.applications import InceptionResNetV2
from tensorflow.keras.losses import sparse_categorical_crossentropy
from tensorflow.keras.optimizers import RMSprop
from sklearn.metrics import classification_report,confusion_matrix
from sklearn.model_selection import train_test_split
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import random
import cv2
import os
```

Image Dataset Import

In [2]:
```
labels = ['1_normal', '2_cataract','3_glaucoma','4_retina_disease']
img_size = 224
def get_data(data_dir):
    data = []

    for label in labels:
        path = os.path.join(data_dir, label)
        class_num = labels.index(label)
        for img in os.listdir(path):
            try:
                img_arr = cv2.imread(os.path.join(path, img))[...,::-1] #convert BGR to RGB format
                crop_image= img_arr[0:1728,430:2190]
                resized_arr = cv2.resize(crop_image, (img_size, img_size)) # Reshaping images to preferred size
                data.append([resized_arr, class_num])
            except Exception as e:
                print(e)
    return np.array(data)
```

In [3]:
```
#function call to get_data function that takes file path of the dataset.
data= get_data('dataset/all_equal_300_images/')
```

```
<ipython-input-2-b08f5e223f84>:17: VisibleDeprecationWarning: Creating an ndarray from ragged nested sequences (which
is a list-or-tuple of lists-or-tuples-or ndarrays with different lengths or shapes) is deprecated. If you meant to do
this, you must specify 'dtype=object' when creating the ndarray
  return np.array(data)
```

In [4]:
```
data.shape
```

Out[4]: (1200, 2)

In [5]:
```
type(data)
```

Out[5]: numpy.ndarray

Dividing Data Ndarray into Normal, Cataract, Glaucoma and Retina diseases.

In [6]:
```
normal= data[0:300]
normal.shape
```

Out[6]: (300, 2)

In [7]:
```
cataract=data[300:600]
cataract.shape
```

Out[7]: (300, 2)

In [8]:
```
glaucoma= data[600:900]
glaucoma.shape
```

Out[8]: (300, 2)

In [9]:
```
retina_disease= data[900:1200]
retina_disease.shape
```

Out[9]:
```
(300, 2)
```

In [10]:
```
random.seed(20)
np.random.shuffle(normal)
np.random.shuffle(cataract)
np.random.shuffle(glaucoma)
np.random.shuffle(retina_disease)
```

Performing Normalization and Resize operation

In [11]:
```
def normalize(x_train,x_val,x_test):

    x_train = np.array(x_train) / 255
    x_train.reshape(-1, img_size, img_size, 1)

    x_test= np.array(x_test) / 255
    x_test.reshape(-1, img_size, img_size, 1)

    x_val= np.array(x_val) / 255
    x_val.reshape(-1, img_size, img_size, 1)

    return (x_train,x_val,x_test)
```

Separating the Images and Labels into Respective Variables

In [12]:
```
def image_label_split(train,validation,test):

    x_train = []
    y_train = []
    x_val = []
    y_val = []
    x_test = []
    y_test = []

    for feature, label in train:
      x_train.append(feature)
      y_train.append(label)

    for feature, label in validation:
      x_val.append(feature)
      y_val.append(label)

    for feature, label in test:
      x_test.append(feature)
      y_test.append(label)

    y_train = np.array(y_train)
    y_val = np.array(y_val)
    y_test= np.array(y_test)

    return (x_train,y_train,x_val,y_val,x_test,y_test)
```

InceptionResNetV2-LSTM MODEL

In [13]:
```
def model_build_compile(k):
    baseModel = InceptionResNetV2(weights="imagenet", include_top=False, input_tensor=Input(shape=(224, 224, 3)))
    for layer in baseModel.layers:
            layer.trainable = False

    x = baseModel.output

        # LSTM layer
    x = Reshape((25, 1536))(x)
    x = ((LSTM(512, activation="relu", return_sequences=True, trainable=False)))(x)
    x = BatchNormalization()(x)
    #

        # FC layer
    x = Flatten(name="flatten")(x)

        # fc1 layer
    x = Dense(units=4096, activation='relu')(x)
    x = BatchNormalization()(x)
    #

        # fc2 layer
    x = Dense(units=4096, activation='relu')(x)
    x = BatchNormalization()(x)
    #

        # Output layer
    output = Dense(units=4, activation='softmax')(x)

    model = Model(inputs=baseModel.input, outputs=output)
    opt = RMSprop(learning_rate=0.01, clipvalue=100)
    model.compile(loss='sparse_categorical_crossentropy', optimizer=opt, metrics=["accuracy"])
    k=k+1
    print("model building and compiling for fold",k)
    return model
```

Model prediction for Test Images and Computation of Sensitivity and Specificity

In [14]:
```
def test_pred(x_val,y_val,k):
    predictions = model.predict(x_val)
    predictions = np.argmax(predictions, axis = -1)

    print('----------------Test accuracy for',k+1,'fold----------------')
    #Confusion matrix, Accuracy, sensitivity and specificity
    cm1 = confusion_matrix(y_val,predictions)
    print('Confusion Matrix : \n', cm1)

    #####from confusion matrix calculate accuracy

    sensitivity_1_normal = (cm1[0,0])/(cm1[0,0]+cm1[0,1]+cm1[0,2]+cm1[0,3])
    #print('Sensitivity_1_normal          : ', sensitivity_1_normal )

    sensitivity_2_cataract = (cm1[1,1])/(cm1[1,0]+cm1[1,1]+cm1[1,2]+cm1[1,3])
    #print('Sensitivity_2_cataract        : ', sensitivity_2_cataract )


    sensitivity_3_glaucoma = (cm1[2,2])/(cm1[2,0]+cm1[2,1]+cm1[2,2]+cm1[2,3])
    #print('Sensitivity_3_glaucoma        : ', sensitivity_3_glaucoma )

    sensitivity_4_retina_disease = (cm1[3,3])/(cm1[3,0]+cm1[3,1]+cm1[3,2]+cm1[3,3])
    #print('Sensitivity_4_retina_disease  : ', sensitivity_4_retina_disease )

    specificity_1_normal = (cm1[1,1]+cm1[1,2]+cm1[1,3]+cm1[2,1]+cm1[2,2]+cm1[2,3]+cm1[3,1]+cm1[3,2]+cm1[3,3])/(cm1[1,0
]+cm1[2,0]+cm1[3,0]+cm1[1,1]+cm1[1,2]+cm1[1,3]+cm1[2,1]+cm1[2,2]+cm1[2,3]+cm1[3,1]+cm1[3,2]+cm1[3,3])
    #print('Specificity : ', specificity_1_normal)

    specificity_2_cataract = (cm1[0,0]+cm1[0,2]+cm1[0,3]+cm1[2,0]+cm1[2,2]+cm1[2,3]+cm1[3,0]+cm1[3,2]+cm1[3,3])/(cm1[0
,1]+cm1[2,1]+cm1[3,1]+cm1[0,0]+cm1[0,2]+cm1[0,3]+cm1[2,0]+cm1[2,2]+cm1[2,3]+cm1[3,0]+cm1[3,2]+cm1[3,3])
    #print('Specificity : ', specificity_2_cataract)

    specificity_3_glaucoma = (cm1[0,0]+cm1[0,1]+cm1[0,3]+cm1[1,0]+cm1[1,1]+cm1[1,3]+cm1[3,0]+cm1[3,1]+cm1[3,3])/(cm1[0
,2]+cm1[1,2]+cm1[3,2]+cm1[0,0]+cm1[0,1]+cm1[0,3]+cm1[1,0]+cm1[1,1]+cm1[1,3]+cm1[3,0]+cm1[3,1]+cm1[3,3])
    #print('Specificity : ', specificity_3_glaucoma)

    specificity_4_retina_disease= (cm1[0,0]+cm1[0,1]+cm1[0,2]+cm1[1,0]+cm1[1,1]+cm1[1,2]+cm1[2,0]+cm1[2,1]+cm1[2,2])/(
cm1[0,3]+cm1[1,3]+cm1[2,3]+cm1[0,0]+cm1[0,1]+cm1[0,2]+cm1[1,0]+cm1[1,1]+cm1[1,2]+cm1[2,0]+cm1[2,1]+cm1[2,2])
    #print('Specificity : ', specificity_4_retina_disease)

    Sensitivity= (sensitivity_1_normal + sensitivity_2_cataract + sensitivity_3_glaucoma + sensitivity_4_retina_diseas
e)/4
    #print(Sensitivity)

    Specificity= (specificity_1_normal + specificity_2_cataract + specificity_3_glaucoma + specificity_4_retina_diseas
e)/4
    #print(Specificity)

    total1=sum(sum(cm1))
    test_accuracy=(cm1[0,0]+cm1[1,1]+cm1[2,2]+cm1[3,3])/total1

    print ('Accuracy    : ', test_accuracy)
    print ('Specificity : ', Specificity)
    print ('Sensitivity : ', Sensitivity)
    print('-----------------------------End of',k+1,'Fold-------------------------')
    return test_accuracy,Specificity,Sensitivity,cm1
```

In [15]:
```
CM= []
test_accuracy=[]
test_sensitivity=[]
test_specificity=[]
train_acc = []
val_acc = []
train_loss = []
val_loss = []
```

InceptionResNetV2-LSTM 5 Fold Cross Validation

In [16]:
```
for k in range (5): # for loop to run 5 folds
    n=30  #specifying the number of images for each class in test phase,calulated as per 10% of total images in each c
lass images 300.


    # Adding the images in normal validation set by using k*n to (k+1)*n as index values for normal dataset divided in
cell 6.
    test_normal= normal[k*n:(k+1)*n]
    print('------------------------Start of',k+1,'Fold------------------------')
    print('test images for normal class from',k*n,(k+1)*n)

    # Adding the images in cataract validation set by using k*n to (k+1)*nas index values for cataract dataset divided
in cell 7.
    test_cataract= cataract[k*n:(k+1)*n]
    print('test images for cataract class from',k*n,(k+1)*n)

    # Adding the images in gluacoma validation set by using k*nto (k+1)*n as index values for gluacoma dataset divided
in cell 8.
    test_glaucoma= glaucoma[k*n:(k+1)*n]
    print('test images for glaucoma class from',k*n,(k+1)*n)

    # Adding the images in retina disease validation set by using k*n to (k+1)*n as index values for retina disease da
taset divided in cell 9.
    test_retina= retina_disease[k*n:(k+1)*n]
    print('test images for retina disease class from',k*n,(k+1)*n)

    # Now for train and validation set of Normal images first adding 0 to k*n images and then adding all the images fr
om (k+1)*n till last image.

    train_validation_normal= normal[:k*n]
    train_validation_normal= np.append(train_validation_normal,normal[(k+1)*n:],axis=0)
    print('train_validation images for normal class from 0 to',k*n,'and',(k+1)*n,'to 300')

    # Now for train and validation set of cataract images first adding 0 to k*n images and then adding all the images
 from (k+1)*n till last image.

    train_validation_cataract= cataract[:k*n]
    train_validation_cataract= np.append(train_validation_cataract,cataract[(k+1)*n:],axis=0)
    print('train_validation images for cataract class from 0 to',k*n,'and',(k+1)*n,'to 300')

    # Now for train and validation set of glaucoma images first adding 0 to k*n images and then adding all the images
 from (k+1)*n till last image.

    train_validation_glaucoma= glaucoma[:k*n]
    train_validation_glaucoma= np.append(train_validation_glaucoma,glaucoma[(k+1)*n:],axis=0)
    print('train_validation  images for glaucoma class from 0',k*n,'and',(k+1)*n,'to 300')

    # Now for train and validation set of retina disease images first adding 0 to k*n images and then adding all the i
mages from (k+1)*n till last image.

    train_validation_retina= retina_disease[:k*n]
    train_validation_retina= np.append(train_validation_retina,retina_disease[(k+1)*n:],axis=0)
    print('train_validation images for retina disease class from 0 to',k*n,'and',(k+1)*n,'to 300')


    # Splitting the train validation datasets in 80:20 ratio which would eventually give us 70% images in train and 2
0% images in validation and 10% in test.

    normal_train, normal_validation                 = train_test_split(train_validation_normal, test_size=0.20, random
_state=14,shuffle=True)
    cataract_train, cataract_validation             = train_test_split(train_validation_cataract, test_size=0.20, rand
om_state=14,shuffle=True)
    glaucoma_train, glaucoma_validation             = train_test_split(train_validation_glaucoma, test_size=0.20, rand
om_state=14,shuffle=True)
    retina_disease_train, retina_disease_validation = train_test_split(train_validation_retina, test_size=0.20, random
_state=14,shuffle=True)

    # Appending all train set images for all classes
    train= np.append(normal_train,cataract_train,axis=0)
    train= np.append(train,glaucoma_train,axis=0)
    train= np.append(train,retina_disease_train,axis=0)

    # Appending all validation set images for all classes
    validation= np.append(normal_validation,cataract_validation,axis=0)
    validation= np.append(validation,glaucoma_validation,axis=0)
    validation= np.append(validation,retina_disease_validation,axis=0)

    # Appending all test set images for all classes
    test= np.append(test_normal,test_cataract,axis=0)
    test= np.append(test,test_glaucoma,axis=0)
    test= np.append(test,test_retina,axis=0)


    # Shuffling the train validation and test set as they are added sequentially.
    random.seed(6)
    np.random.shuffle(train)
    np.random.shuffle(validation)
    np.random.shuffle(test)
```

```
    # Passing the train validation test as argument for image_label_split function that return features and labels sep
arated.
    x_train,y_train,x_val,y_val,x_test,y_test = image_label_split(train,validation,test)

    # Passing the x_Train x_val and x_test as a argument for normalize function that returns the normalized and reshap
ed sets.
    x_train,x_val,x_test = normalize(x_train,x_val,x_test)

    # model building and model compile is done using a model_build_compile().

    model = model_build_compile(k)
    history = model.fit(x_train,y_train,epochs =50, validation_data = (x_val,y_val))


    train_acc = np.append(train_acc,history.history['accuracy'])
    val_acc = np.append(val_acc,history.history['val_accuracy'])

    train_loss = np.append(train_loss,history.history['loss'])
    val_loss = np.append(val_loss,history.history['val_loss'])

    x,y,z,c = test_pred(x_test,y_test,k)


    CM.append([c])
    test_accuracy.append(x)
    test_specificity.append(y)
    test_sensitivity.append(z)
```

```
------------------------Start of 1 Fold------------------------
test images for normal class from 0 30
test images for cataract class from 0 30
test images for glaucoma class from 0 30
test images for retina disease class from 0 30
train_validation images for normal class from 0 to 0 and 30 to 300
train_validation images for cataract class from 0 to 0 and 30 to 300
train_validation  images for glaucoma class from 0 0 and 30 to 300
train_validation images for retina disease class from 0 to 0 and 30 to 300
model building and compiling for fold 1
Epoch 1/50
27/27 [==============================] - 117s 4s/step - loss: 14.0160 - accuracy: 0.4282 - val_loss: 35.8210 - val_ac
curacy: 0.4583
Epoch 2/50
27/27 [==============================] - 113s 4s/step - loss: 5.9905 - accuracy: 0.5301 - val_loss: 6.7669 - val_accu
racy: 0.4815
Epoch 3/50
27/27 [==============================] - 119s 4s/step - loss: 3.4671 - accuracy: 0.6019 - val_loss: 3.3664 - val_accu
racy: 0.4630
Epoch 4/50
27/27 [==============================] - 124s 5s/step - loss: 1.8882 - accuracy: 0.6308 - val_loss: 4.1670 - val_accu
racy: 0.4630
Epoch 5/50
27/27 [==============================] - 124s 5s/step - loss: 1.4832 - accuracy: 0.7106 - val_loss: 2.8696 - val_accu
racy: 0.4491
Epoch 6/50
27/27 [==============================] - 124s 5s/step - loss: 1.5334 - accuracy: 0.7141 - val_loss: 2.3905 - val_accu
racy: 0.4861
Epoch 7/50
27/27 [==============================] - 122s 5s/step - loss: 1.0793 - accuracy: 0.7743 - val_loss: 3.2301 - val_accu
racy: 0.4028
Epoch 8/50
27/27 [==============================] - 123s 5s/step - loss: 0.8415 - accuracy: 0.8229 - val_loss: 3.0803 - val_accu
racy: 0.5509
Epoch 9/50
27/27 [==============================] - 125s 5s/step - loss: 0.8222 - accuracy: 0.8009 - val_loss: 4.7645 - val_accu
racy: 0.5139
Epoch 10/50
27/27 [==============================] - 130s 5s/step - loss: 0.5144 - accuracy: 0.8866 - val_loss: 6.7335 - val_accu
racy: 0.4398
Epoch 11/50
27/27 [==============================] - 133s 5s/step - loss: 0.3941 - accuracy: 0.9062 - val_loss: 6.0655 - val_accu
racy: 0.5000
Epoch 12/50
27/27 [==============================] - 127s 5s/step - loss: 0.1897 - accuracy: 0.9398 - val_loss: 4.3674 - val_accu
racy: 0.5278
Epoch 13/50
27/27 [==============================] - 132s 5s/step - loss: 0.3329 - accuracy: 0.9086 - val_loss: 1.3707 - val_accu
racy: 0.7593
Epoch 14/50
27/27 [==============================] - 129s 5s/step - loss: 0.2042 - accuracy: 0.9387 - val_loss: 3.3888 - val_accu
racy: 0.5278
Epoch 15/50
27/27 [==============================] - 133s 5s/step - loss: 0.1716 - accuracy: 0.9421 - val_loss: 2.6805 - val_accu
racy: 0.6204
Epoch 16/50
27/27 [==============================] - 134s 5s/step - loss: 0.2058 - accuracy: 0.9387 - val_loss: 2.3668 - val_accu
racy: 0.7130
Epoch 17/50
27/27 [==============================] - 131s 5s/step - loss: 0.1704 - accuracy: 0.9606 - val_loss: 2.5437 - val_accu
racy: 0.6296
Epoch 18/50
27/27 [==============================] - 131s 5s/step - loss: 0.2543 - accuracy: 0.9433 - val_loss: 3.0498 - val_accu
racy: 0.6111
Epoch 19/50
27/27 [==============================] - 129s 5s/step - loss: 0.1108 - accuracy: 0.9699 - val_loss: 3.2019 - val_accu
racy: 0.5880
Epoch 20/50
27/27 [==============================] - 125s 5s/step - loss: 0.1539 - accuracy: 0.9722 - val_loss: 4.1400 - val_accu
racy: 0.5926
Epoch 21/50
27/27 [==============================] - 136s 5s/step - loss: 0.0964 - accuracy: 0.9664 - val_loss: 4.1560 - val_accu
racy: 0.5417
Epoch 22/50
27/27 [==============================] - 126s 5s/step - loss: 0.1637 - accuracy: 0.9630 - val_loss: 3.7231 - val_accu
racy: 0.6204
Epoch 23/50
27/27 [==============================] - 130s 5s/step - loss: 0.1488 - accuracy: 0.9653 - val_loss: 3.8131 - val_accu
racy: 0.6204
Epoch 24/50
27/27 [==============================] - 128s 5s/step - loss: 0.1709 - accuracy: 0.9630 - val_loss: 3.5829 - val_accu
racy: 0.6991
Epoch 25/50
27/27 [==============================] - 128s 5s/step - loss: 0.1424 - accuracy: 0.9699 - val_loss: 2.9909 - val_accu
racy: 0.6852
Epoch 26/50
27/27 [==============================] - 123s 5s/step - loss: 0.1820 - accuracy: 0.9549 - val_loss: 4.6910 - val_accu
racy: 0.6481
```

```
Epoch 27/50
27/27 [==============================] - 123s 5s/step - loss: 0.0716 - accuracy: 0.9826 - val_loss: 2.9320 - val_accu
racy: 0.6944
Epoch 28/50
27/27 [==============================] - 123s 5s/step - loss: 0.0961 - accuracy: 0.9769 - val_loss: 3.1959 - val_accu
racy: 0.6620
Epoch 29/50
27/27 [==============================] - 123s 5s/step - loss: 0.0380 - accuracy: 0.9907 - val_loss: 3.0136 - val_accu
racy: 0.7222
Epoch 30/50
27/27 [==============================] - 123s 5s/step - loss: 0.1406 - accuracy: 0.9688 - val_loss: 3.5050 - val_accu
racy: 0.7130
Epoch 31/50
27/27 [==============================] - 123s 5s/step - loss: 0.1158 - accuracy: 0.9699 - val_loss: 5.0975 - val_accu
racy: 0.6528
Epoch 32/50
27/27 [==============================] - 122s 5s/step - loss: 0.0865 - accuracy: 0.9745 - val_loss: 105.4086 - val_ac
curacy: 0.5648
Epoch 33/50
27/27 [==============================] - 123s 5s/step - loss: 0.1439 - accuracy: 0.9826 - val_loss: 3.8572 - val_accu
racy: 0.6620
Epoch 34/50
27/27 [==============================] - 124s 5s/step - loss: 0.1207 - accuracy: 0.9711 - val_loss: 5.7280 - val_accu
racy: 0.6389
Epoch 35/50
27/27 [==============================] - 122s 5s/step - loss: 0.0475 - accuracy: 0.9850 - val_loss: 13.4978 - val_acc
uracy: 0.6991
Epoch 36/50
27/27 [==============================] - 122s 5s/step - loss: 0.0560 - accuracy: 0.9826 - val_loss: 79.6286 - val_acc
uracy: 0.6065
Epoch 37/50
27/27 [==============================] - 122s 5s/step - loss: 0.3147 - accuracy: 0.9769 - val_loss: 3.5545 - val_accu
racy: 0.7130
Epoch 38/50
27/27 [==============================] - 122s 5s/step - loss: 0.0847 - accuracy: 0.9803 - val_loss: 3.2420 - val_accu
racy: 0.7130
Epoch 39/50
27/27 [==============================] - 122s 5s/step - loss: 0.0559 - accuracy: 0.9850 - val_loss: 3.9128 - val_accu
racy: 0.7130
Epoch 40/50
27/27 [==============================] - 122s 5s/step - loss: 0.0343 - accuracy: 0.9884 - val_loss: 3.6491 - val_accu
racy: 0.7222
Epoch 41/50
27/27 [==============================] - 126s 5s/step - loss: 0.0714 - accuracy: 0.9850 - val_loss: 3.0346 - val_accu
racy: 0.7130
Epoch 42/50
27/27 [==============================] - 124s 5s/step - loss: 0.0454 - accuracy: 0.9873 - val_loss: 3.2258 - val_accu
racy: 0.6991
Epoch 43/50
27/27 [==============================] - 129s 5s/step - loss: 0.1222 - accuracy: 0.9780 - val_loss: 28.5134 - val_acc
uracy: 0.5926
Epoch 44/50
27/27 [==============================] - 129s 5s/step - loss: 0.2885 - accuracy: 0.9688 - val_loss: 4.8965 - val_accu
racy: 0.6389
Epoch 45/50
27/27 [==============================] - 135s 5s/step - loss: 0.1764 - accuracy: 0.9838 - val_loss: 4.8542 - val_accu
racy: 0.6944
Epoch 46/50
27/27 [==============================] - 127s 5s/step - loss: 0.0872 - accuracy: 0.9850 - val_loss: 3.6788 - val_accu
racy: 0.6898
Epoch 47/50
27/27 [==============================] - 130s 5s/step - loss: 0.0468 - accuracy: 0.9919 - val_loss: 3.5965 - val_accu
racy: 0.6991
Epoch 48/50
27/27 [==============================] - 137s 5s/step - loss: 0.0584 - accuracy: 0.9850 - val_loss: 3.7078 - val_accu
racy: 0.6806
Epoch 49/50
27/27 [==============================] - 136s 5s/step - loss: 0.1302 - accuracy: 0.9722 - val_loss: 3.4049 - val_accu
racy: 0.7083
Epoch 50/50
27/27 [==============================] - 130s 5s/step - loss: 0.0958 - accuracy: 0.9873 - val_loss: 4.5309 - val_accu
racy: 0.6898
----------------Test accuracy for 1 fold----------------
Confusion Matrix :
 [[19  0  8  3]
 [ 5 18  4  3]
 [ 2  0 26  2]
 [ 7  1 10 12]]
Accuracy    :  0.625
Specificity :  0.8400558523555124
Sensitivity :  0.625
-----------------------------End of 1 Fold--------------------------
------------------------Start of 2 Fold-----------------------
test images for normal class from 30 60
test images for cataract class from 30 60
test images for glaucoma class from 30 60
test images for retina disease class from 30 60
train_validation images for normal class from 0 to 30 and 60 to 300
```

```
train_validation images for cataract class from 0 to 30 and 60 to 300
train_validation  images for glaucoma class from 0 30 and 60 to 300
train_validation images for retina disease class from 0 to 30 and 60 to 300
model building and compiling for fold 2
Epoch 1/50
27/27 [==============================] - 139s 5s/step - loss: 12.7534 - accuracy: 0.4259 - val_loss: 20.9874 - val_ac
curacy: 0.4630
Epoch 2/50
27/27 [==============================] - 125s 5s/step - loss: 4.2155 - accuracy: 0.5498 - val_loss: 6.2494 - val_accu
racy: 0.5093
Epoch 3/50
27/27 [==============================] - 140s 5s/step - loss: 2.2095 - accuracy: 0.5949 - val_loss: 1.8864 - val_accu
racy: 0.6065
Epoch 4/50
27/27 [==============================] - 136s 5s/step - loss: 2.1300 - accuracy: 0.6227 - val_loss: 3.5264 - val_accu
racy: 0.4537
Epoch 5/50
27/27 [==============================] - 129s 5s/step - loss: 1.5662 - accuracy: 0.7060 - val_loss: 2.7486 - val_accu
racy: 0.4954
Epoch 6/50
27/27 [==============================] - 133s 5s/step - loss: 1.1008 - accuracy: 0.7569 - val_loss: 1.9878 - val_accu
racy: 0.5417
Epoch 7/50
27/27 [==============================] - 129s 5s/step - loss: 1.0654 - accuracy: 0.8056 - val_loss: 5.1012 - val_accu
racy: 0.4907
Epoch 8/50
27/27 [==============================] - 133s 5s/step - loss: 0.7922 - accuracy: 0.8519 - val_loss: 2.9329 - val_accu
racy: 0.5509
Epoch 9/50
27/27 [==============================] - 136s 5s/step - loss: 0.7478 - accuracy: 0.8519 - val_loss: 2.7076 - val_accu
racy: 0.6111
Epoch 10/50
27/27 [==============================] - 136s 5s/step - loss: 0.5467 - accuracy: 0.8657 - val_loss: 4.8380 - val_accu
racy: 0.5370
Epoch 11/50
27/27 [==============================] - 136s 5s/step - loss: 0.4135 - accuracy: 0.9190 - val_loss: 5.5357 - val_accu
racy: 0.4352
Epoch 12/50
27/27 [==============================] - 134s 5s/step - loss: 0.3940 - accuracy: 0.9051 - val_loss: 4.4413 - val_accu
racy: 0.5602
Epoch 13/50
27/27 [==============================] - 135s 5s/step - loss: 0.2160 - accuracy: 0.9375 - val_loss: 5.4398 - val_accu
racy: 0.4907
Epoch 14/50
27/27 [==============================] - 134s 5s/step - loss: 0.6088 - accuracy: 0.8947 - val_loss: 2.9691 - val_accu
racy: 0.6296
Epoch 15/50
27/27 [==============================] - 137s 5s/step - loss: 0.2804 - accuracy: 0.9398 - val_loss: 2.7835 - val_accu
racy: 0.6713
Epoch 16/50
27/27 [==============================] - 130s 5s/step - loss: 0.2239 - accuracy: 0.9468 - val_loss: 3.2810 - val_accu
racy: 0.7037
Epoch 17/50
27/27 [==============================] - 128s 5s/step - loss: 0.2480 - accuracy: 0.9340 - val_loss: 2.6604 - val_accu
racy: 0.6991
Epoch 18/50
27/27 [==============================] - 126s 5s/step - loss: 0.1392 - accuracy: 0.9572 - val_loss: 3.3575 - val_accu
racy: 0.6157
Epoch 19/50
27/27 [==============================] - 127s 5s/step - loss: 0.1725 - accuracy: 0.9711 - val_loss: 4.7651 - val_accu
racy: 0.6065
Epoch 20/50
27/27 [==============================] - 128s 5s/step - loss: 0.2810 - accuracy: 0.9259 - val_loss: 3.5301 - val_accu
racy: 0.6620
Epoch 21/50
27/27 [==============================] - 133s 5s/step - loss: 0.1361 - accuracy: 0.9630 - val_loss: 2.4522 - val_accu
racy: 0.7685
Epoch 22/50
27/27 [==============================] - 126s 5s/step - loss: 0.2499 - accuracy: 0.9815 - val_loss: 4.2910 - val_accu
racy: 0.7083
Epoch 23/50
27/27 [==============================] - 126s 5s/step - loss: 0.0938 - accuracy: 0.9769 - val_loss: 3.4247 - val_accu
racy: 0.7176
Epoch 24/50
27/27 [==============================] - 132s 5s/step - loss: 0.0987 - accuracy: 0.9757 - val_loss: 3.6399 - val_accu
racy: 0.6898
Epoch 25/50
27/27 [==============================] - 129s 5s/step - loss: 0.3806 - accuracy: 0.9676 - val_loss: 5.7035 - val_accu
racy: 0.6065
Epoch 26/50
27/27 [==============================] - 128s 5s/step - loss: 0.1539 - accuracy: 0.9491 - val_loss: 4.2075 - val_accu
racy: 0.6620
Epoch 27/50
27/27 [==============================] - 132s 5s/step - loss: 0.3406 - accuracy: 0.9572 - val_loss: 3.2580 - val_accu
racy: 0.7083
Epoch 28/50
27/27 [==============================] - 140s 5s/step - loss: 0.1400 - accuracy: 0.9722 - val_loss: 2.6562 - val_accu
racy: 0.7731
```

```
Epoch 29/50
27/27 [==============================] - 142s 5s/step - loss: 0.1750 - accuracy: 0.9664 - val_loss: 3.2005 - val_accu
racy: 0.7222
Epoch 30/50
27/27 [==============================] - 135s 5s/step - loss: 0.1865 - accuracy: 0.9769 - val_loss: 3.9516 - val_accu
racy: 0.6528
Epoch 31/50
27/27 [==============================] - 127s 5s/step - loss: 0.0838 - accuracy: 0.9792 - val_loss: 3.8573 - val_accu
racy: 0.6991
Epoch 32/50
27/27 [==============================] - 131s 5s/step - loss: 0.1012 - accuracy: 0.9803 - val_loss: 5.1107 - val_accu
racy: 0.5602
Epoch 33/50
27/27 [==============================] - 133s 5s/step - loss: 0.2406 - accuracy: 0.9583 - val_loss: 6.3935 - val_accu
racy: 0.5463
Epoch 34/50
27/27 [==============================] - 122s 5s/step - loss: 0.0810 - accuracy: 0.9803 - val_loss: 3.6292 - val_accu
racy: 0.6991
Epoch 35/50
27/27 [==============================] - 123s 5s/step - loss: 0.0386 - accuracy: 0.9884 - val_loss: 4.7817 - val_accu
racy: 0.7083
Epoch 36/50
27/27 [==============================] - 122s 5s/step - loss: 0.0986 - accuracy: 0.9722 - val_loss: 3.5633 - val_accu
racy: 0.6991
Epoch 37/50
27/27 [==============================] - 121s 5s/step - loss: 0.0752 - accuracy: 0.9850 - val_loss: 2.4594 - val_accu
racy: 0.7731
Epoch 38/50
27/27 [==============================] - 121s 5s/step - loss: 0.0523 - accuracy: 0.9896 - val_loss: 3.5475 - val_accu
racy: 0.7361
Epoch 39/50
27/27 [==============================] - 128s 5s/step - loss: 0.1448 - accuracy: 0.9676 - val_loss: 3.3004 - val_accu
racy: 0.7222
Epoch 40/50
27/27 [==============================] - 123s 5s/step - loss: 0.0621 - accuracy: 0.9838 - val_loss: 3.0746 - val_accu
racy: 0.7269
Epoch 41/50
27/27 [==============================] - 129s 5s/step - loss: 0.0324 - accuracy: 0.9896 - val_loss: 2.9246 - val_accu
racy: 0.7407
Epoch 42/50
27/27 [==============================] - 122s 5s/step - loss: 0.1763 - accuracy: 0.9792 - val_loss: 2.7042 - val_accu
racy: 0.7407
Epoch 43/50
27/27 [==============================] - 123s 5s/step - loss: 0.0745 - accuracy: 0.9873 - val_loss: 9.0967 - val_accu
racy: 0.5602
Epoch 44/50
27/27 [==============================] - 123s 5s/step - loss: 0.0397 - accuracy: 0.9884 - val_loss: 7.8803 - val_accu
racy: 0.6435
Epoch 45/50
27/27 [==============================] - 122s 5s/step - loss: 0.0619 - accuracy: 0.9861 - val_loss: 7.8349 - val_accu
racy: 0.7037
Epoch 46/50
27/27 [==============================] - 122s 5s/step - loss: 0.0374 - accuracy: 0.9896 - val_loss: 4.8230 - val_accu
racy: 0.7222
Epoch 47/50
27/27 [==============================] - 123s 5s/step - loss: 0.0449 - accuracy: 0.9873 - val_loss: 4.2222 - val_accu
racy: 0.7176
Epoch 48/50
27/27 [==============================] - 121s 5s/step - loss: 0.0563 - accuracy: 0.9873 - val_loss: 4.0481 - val_accu
racy: 0.6944
Epoch 49/50
27/27 [==============================] - 121s 5s/step - loss: 0.0532 - accuracy: 0.9931 - val_loss: 5.5798 - val_accu
racy: 0.7269
Epoch 50/50
27/27 [==============================] - 122s 5s/step - loss: 0.1154 - accuracy: 0.9792 - val_loss: 3.3464 - val_accu
racy: 0.7593
----------------Test accuracy for 2 fold----------------
Confusion Matrix :
 [[26  0  2  2]
 [ 5 20  4  1]
 [ 8  2 16  4]
 [ 6  1  4 19]]
Accuracy    :  0.675
Specificity :  0.8653964086368978
Sensitivity :  0.6749999999999999
----------------------------End of 2 Fold-------------------------
-----------------------Start of 3 Fold-----------------------
test images for normal class from 60 90
test images for cataract class from 60 90
test images for glaucoma class from 60 90
test images for retina disease class from 60 90
train_validation images for normal class from 0 to 60 and 90 to 300
train_validation images for cataract class from 0 to 60 and 90 to 300
train_validation  images for glaucoma class from 0 60 and 90 to 300
train_validation images for retina disease class from 0 to 60 and 90 to 300
model building and compiling for fold 3
Epoch 1/50
27/27 [==============================] - 137s 5s/step - loss: 13.8976 - accuracy: 0.4433 - val_loss: 35.6255 - val_ac
```

```
curacy: 0.4352
Epoch 2/50
27/27 [==============================] - 123s 5s/step - loss: 5.0203 - accuracy: 0.5231 - val_loss: 9.3185 - val_accu
racy: 0.5370
Epoch 3/50
27/27 [==============================] - 125s 5s/step - loss: 4.6516 - accuracy: 0.5706 - val_loss: 6.2850 - val_accu
racy: 0.5509
Epoch 4/50
27/27 [==============================] - 123s 5s/step - loss: 2.0317 - accuracy: 0.5995 - val_loss: 3.4893 - val_accu
racy: 0.4491
Epoch 5/50
27/27 [==============================] - 124s 5s/step - loss: 1.6489 - accuracy: 0.6759 - val_loss: 2.6440 - val_accu
racy: 0.5556
Epoch 6/50
27/27 [==============================] - 124s 5s/step - loss: 1.3643 - accuracy: 0.7500 - val_loss: 6.2596 - val_accu
racy: 0.4630
Epoch 7/50
27/27 [==============================] - 126s 5s/step - loss: 1.6035 - accuracy: 0.7234 - val_loss: 6.8552 - val_accu
racy: 0.3472
Epoch 8/50
27/27 [==============================] - 125s 5s/step - loss: 1.1673 - accuracy: 0.7986 - val_loss: 4.1814 - val_accu
racy: 0.5185
Epoch 9/50
27/27 [==============================] - 128s 5s/step - loss: 0.6317 - accuracy: 0.8356 - val_loss: 3.7405 - val_accu
racy: 0.4954
Epoch 10/50
27/27 [==============================] - 127s 5s/step - loss: 0.7421 - accuracy: 0.8576 - val_loss: 3.1667 - val_accu
racy: 0.5000
Epoch 11/50
27/27 [==============================] - 128s 5s/step - loss: 0.6489 - accuracy: 0.8669 - val_loss: 2.6748 - val_accu
racy: 0.6111
Epoch 12/50
27/27 [==============================] - 124s 5s/step - loss: 0.4206 - accuracy: 0.9005 - val_loss: 3.4233 - val_accu
racy: 0.5833
Epoch 13/50
27/27 [==============================] - 124s 5s/step - loss: 0.5461 - accuracy: 0.9120 - val_loss: 1.8846 - val_accu
racy: 0.7130
Epoch 14/50
27/27 [==============================] - 125s 5s/step - loss: 0.3769 - accuracy: 0.9225 - val_loss: 2.8898 - val_accu
racy: 0.6065
Epoch 15/50
27/27 [==============================] - 130s 5s/step - loss: 0.5059 - accuracy: 0.9120 - val_loss: 8.5519 - val_accu
racy: 0.4676
Epoch 16/50
27/27 [==============================] - 130s 5s/step - loss: 0.1967 - accuracy: 0.9502 - val_loss: 2.6525 - val_accu
racy: 0.6944
Epoch 17/50
27/27 [==============================] - 125s 5s/step - loss: 0.2671 - accuracy: 0.9329 - val_loss: 3.8300 - val_accu
racy: 0.6759
Epoch 18/50
27/27 [==============================] - 122s 5s/step - loss: 0.1605 - accuracy: 0.9525 - val_loss: 3.4273 - val_accu
racy: 0.6481
Epoch 19/50
27/27 [==============================] - 130s 5s/step - loss: 0.2723 - accuracy: 0.9410 - val_loss: 2.3562 - val_accu
racy: 0.6944
Epoch 20/50
27/27 [==============================] - 126s 5s/step - loss: 0.1939 - accuracy: 0.9699 - val_loss: 3.4210 - val_accu
racy: 0.6435
Epoch 21/50
27/27 [==============================] - 126s 5s/step - loss: 0.1276 - accuracy: 0.9699 - val_loss: 3.3968 - val_accu
racy: 0.6250
Epoch 22/50
27/27 [==============================] - 124s 5s/step - loss: 0.1685 - accuracy: 0.9572 - val_loss: 3.2326 - val_accu
racy: 0.6667
Epoch 23/50
27/27 [==============================] - 124s 5s/step - loss: 0.2216 - accuracy: 0.9491 - val_loss: 3.6113 - val_accu
racy: 0.6667
Epoch 24/50
27/27 [==============================] - 123s 5s/step - loss: 0.0948 - accuracy: 0.9769 - val_loss: 3.4690 - val_accu
racy: 0.6296
Epoch 25/50
27/27 [==============================] - 123s 5s/step - loss: 0.1498 - accuracy: 0.9688 - val_loss: 2.9071 - val_accu
racy: 0.6806
Epoch 26/50
27/27 [==============================] - 116s 4s/step - loss: 0.2061 - accuracy: 0.9583 - val_loss: 4.2497 - val_accu
racy: 0.6574
Epoch 27/50
27/27 [==============================] - 117s 4s/step - loss: 0.1648 - accuracy: 0.9711 - val_loss: 5.7189 - val_accu
racy: 0.6296
Epoch 28/50
27/27 [==============================] - 123s 5s/step - loss: 0.1079 - accuracy: 0.9734 - val_loss: 4.3305 - val_accu
racy: 0.6389
Epoch 29/50
27/27 [==============================] - 116s 4s/step - loss: 0.0987 - accuracy: 0.9676 - val_loss: 4.4512 - val_accu
racy: 0.6667
Epoch 30/50
27/27 [==============================] - 117s 4s/step - loss: 0.1274 - accuracy: 0.9711 - val_loss: 5.3096 - val_accu
racy: 0.6343
```

```
Epoch 31/50
27/27 [==============================] - 123s 5s/step - loss: 0.0892 - accuracy: 0.9711 - val_loss: 4.1369 - val_accu
racy: 0.6574
Epoch 32/50
27/27 [==============================] - 122s 5s/step - loss: 0.0801 - accuracy: 0.9769 - val_loss: 2.6271 - val_accu
racy: 0.7315
Epoch 33/50
27/27 [==============================] - 124s 5s/step - loss: 0.1111 - accuracy: 0.9757 - val_loss: 4.2594 - val_accu
racy: 0.6620
Epoch 34/50
27/27 [==============================] - 117s 4s/step - loss: 0.0311 - accuracy: 0.9942 - val_loss: 4.3196 - val_accu
racy: 0.6898
Epoch 35/50
27/27 [==============================] - 120s 4s/step - loss: 0.1550 - accuracy: 0.9711 - val_loss: 2.9815 - val_accu
racy: 0.6944
Epoch 36/50
27/27 [==============================] - 121s 5s/step - loss: 0.1042 - accuracy: 0.9711 - val_loss: 5.4110 - val_accu
racy: 0.6296
Epoch 37/50
27/27 [==============================] - 121s 5s/step - loss: 0.0612 - accuracy: 0.9792 - val_loss: 3.4154 - val_accu
racy: 0.6944
Epoch 38/50
27/27 [==============================] - 119s 4s/step - loss: 0.0732 - accuracy: 0.9769 - val_loss: 4.9899 - val_accu
racy: 0.6389
Epoch 39/50
27/27 [==============================] - 115s 4s/step - loss: 0.0674 - accuracy: 0.9803 - val_loss: 3.8290 - val_accu
racy: 0.7037
Epoch 40/50
27/27 [==============================] - 123s 5s/step - loss: 0.0308 - accuracy: 0.9919 - val_loss: 4.3083 - val_accu
racy: 0.6898
Epoch 41/50
27/27 [==============================] - 116s 4s/step - loss: 0.0953 - accuracy: 0.9815 - val_loss: 3.4801 - val_accu
racy: 0.6944
Epoch 42/50
27/27 [==============================] - 120s 4s/step - loss: 0.1120 - accuracy: 0.9769 - val_loss: 5.0744 - val_accu
racy: 0.6019
Epoch 43/50
27/27 [==============================] - 119s 4s/step - loss: 0.0637 - accuracy: 0.9757 - val_loss: 4.7038 - val_accu
racy: 0.6759
Epoch 44/50
27/27 [==============================] - 114s 4s/step - loss: 0.1149 - accuracy: 0.9769 - val_loss: 4.1668 - val_accu
racy: 0.6713
Epoch 45/50
27/27 [==============================] - 126s 5s/step - loss: 0.0293 - accuracy: 0.9884 - val_loss: 5.8880 - val_accu
racy: 0.6481
Epoch 46/50
27/27 [==============================] - 118s 4s/step - loss: 0.0354 - accuracy: 0.9896 - val_loss: 5.0818 - val_accu
racy: 0.6898
Epoch 47/50
27/27 [==============================] - 116s 4s/step - loss: 0.0503 - accuracy: 0.9884 - val_loss: 7.5691 - val_accu
racy: 0.6898
Epoch 48/50
27/27 [==============================] - 123s 5s/step - loss: 0.0495 - accuracy: 0.9873 - val_loss: 6.2970 - val_accu
racy: 0.6157
Epoch 49/50
27/27 [==============================] - 122s 5s/step - loss: 0.0393 - accuracy: 0.9919 - val_loss: 3.2327 - val_accu
racy: 0.6944
Epoch 50/50
27/27 [==============================] - 116s 4s/step - loss: 0.1176 - accuracy: 0.9757 - val_loss: 3.8983 - val_accu
racy: 0.6713
----------------Test accuracy for 3 fold----------------
Confusion Matrix :
 [[13  1  2 14]
 [ 1 25  0  4]
 [ 1  6  7 16]
 [ 4  3  0 23]]
Accuracy    :  0.5666666666666667
Specificity :  0.8127085800994504
Sensitivity :  0.5666666666666667
----------------------------End of 3 Fold-------------------------
-----------------------Start of 4 Fold-----------------------
test images for normal class from 90 120
test images for cataract class from 90 120
test images for glaucoma class from 90 120
test images for retina disease class from 90 120
train_validation images for normal class from 0 to 90 and 120 to 300
train_validation images for cataract class from 0 to 90 and 120 to 300
train_validation  images for glaucoma class from 0 90 and 120 to 300
train_validation images for retina disease class from 0 to 90 and 120 to 300
model building and compiling for fold 4
Epoch 1/50
27/27 [==============================] - 139s 5s/step - loss: 13.8217 - accuracy: 0.4213 - val_loss: 35.2634 - val_ac
curacy: 0.4907
Epoch 2/50
27/27 [==============================] - 118s 4s/step - loss: 5.5966 - accuracy: 0.5336 - val_loss: 8.5264 - val_accu
racy: 0.4213
Epoch 3/50
27/27 [==============================] - 126s 5s/step - loss: 3.3303 - accuracy: 0.5625 - val_loss: 3.0666 - val_accu
```

```
racy: 0.6343
Epoch 4/50
27/27 [==============================] - 125s 5s/step - loss: 2.5186 - accuracy: 0.6215 - val_loss: 2.1875 - val_accu
racy: 0.5278
Epoch 5/50
27/27 [==============================] - 126s 5s/step - loss: 2.0026 - accuracy: 0.7060 - val_loss: 3.7732 - val_accu
racy: 0.4537
Epoch 6/50
27/27 [==============================] - 127s 5s/step - loss: 1.4304 - accuracy: 0.7106 - val_loss: 5.9470 - val_accu
racy: 0.3981
Epoch 7/50
27/27 [==============================] - 133s 5s/step - loss: 1.2151 - accuracy: 0.7535 - val_loss: 5.8028 - val_accu
racy: 0.4583
Epoch 8/50
27/27 [==============================] - 131s 5s/step - loss: 0.7249 - accuracy: 0.8414 - val_loss: 3.3048 - val_accu
racy: 0.5556
Epoch 9/50
27/27 [==============================] - 132s 5s/step - loss: 0.7538 - accuracy: 0.8183 - val_loss: 5.6883 - val_accu
racy: 0.4213
Epoch 10/50
27/27 [==============================] - 127s 5s/step - loss: 0.5473 - accuracy: 0.8657 - val_loss: 2.7573 - val_accu
racy: 0.6250
Epoch 11/50
27/27 [==============================] - 129s 5s/step - loss: 0.6234 - accuracy: 0.8762 - val_loss: 2.3624 - val_accu
racy: 0.6019
Epoch 12/50
27/27 [==============================] - 129s 5s/step - loss: 0.4347 - accuracy: 0.9016 - val_loss: 2.1487 - val_accu
racy: 0.5463
Epoch 13/50
27/27 [==============================] - 129s 5s/step - loss: 0.3393 - accuracy: 0.9074 - val_loss: 1.6821 - val_accu
racy: 0.6852
Epoch 14/50
27/27 [==============================] - 133s 5s/step - loss: 0.2199 - accuracy: 0.9433 - val_loss: 1.5024 - val_accu
racy: 0.7361
Epoch 15/50
27/27 [==============================] - 130s 5s/step - loss: 0.2976 - accuracy: 0.9259 - val_loss: 3.1856 - val_accu
racy: 0.6250
Epoch 16/50
27/27 [==============================] - 129s 5s/step - loss: 0.2163 - accuracy: 0.9421 - val_loss: 3.0222 - val_accu
racy: 0.5602
Epoch 17/50
27/27 [==============================] - 128s 5s/step - loss: 0.2327 - accuracy: 0.9456 - val_loss: 3.1310 - val_accu
racy: 0.6435
Epoch 18/50
27/27 [==============================] - 131s 5s/step - loss: 0.1677 - accuracy: 0.9560 - val_loss: 2.5301 - val_accu
racy: 0.6528
Epoch 19/50
27/27 [==============================] - 131s 5s/step - loss: 0.1609 - accuracy: 0.9502 - val_loss: 2.3286 - val_accu
racy: 0.7315
Epoch 20/50
27/27 [==============================] - 128s 5s/step - loss: 0.1571 - accuracy: 0.9641 - val_loss: 2.0490 - val_accu
racy: 0.6944
Epoch 21/50
27/27 [==============================] - 129s 5s/step - loss: 0.0528 - accuracy: 0.9815 - val_loss: 2.2139 - val_accu
racy: 0.6806
Epoch 22/50
27/27 [==============================] - 128s 5s/step - loss: 0.1170 - accuracy: 0.9664 - val_loss: 2.4887 - val_accu
racy: 0.6852
Epoch 23/50
27/27 [==============================] - 129s 5s/step - loss: 0.1419 - accuracy: 0.9653 - val_loss: 2.1851 - val_accu
racy: 0.6574
Epoch 24/50
27/27 [==============================] - 129s 5s/step - loss: 0.0861 - accuracy: 0.9734 - val_loss: 2.6633 - val_accu
racy: 0.7037
Epoch 25/50
27/27 [==============================] - 129s 5s/step - loss: 0.0868 - accuracy: 0.9653 - val_loss: 2.4715 - val_accu
racy: 0.6713
Epoch 26/50
27/27 [==============================] - 128s 5s/step - loss: 0.1059 - accuracy: 0.9745 - val_loss: 2.6061 - val_accu
racy: 0.7083
Epoch 27/50
27/27 [==============================] - 128s 5s/step - loss: 0.1139 - accuracy: 0.9757 - val_loss: 2.4939 - val_accu
racy: 0.6528
Epoch 28/50
27/27 [==============================] - 129s 5s/step - loss: 0.0816 - accuracy: 0.9780 - val_loss: 2.1807 - val_accu
racy: 0.7269
Epoch 29/50
27/27 [==============================] - 129s 5s/step - loss: 0.1446 - accuracy: 0.9711 - val_loss: 2.8201 - val_accu
racy: 0.7176
Epoch 30/50
27/27 [==============================] - 129s 5s/step - loss: 0.1194 - accuracy: 0.9722 - val_loss: 2.7861 - val_accu
racy: 0.6944
Epoch 31/50
27/27 [==============================] - 129s 5s/step - loss: 0.0665 - accuracy: 0.9803 - val_loss: 2.5696 - val_accu
racy: 0.7130
Epoch 32/50
27/27 [==============================] - 130s 5s/step - loss: 0.1180 - accuracy: 0.9745 - val_loss: 2.1480 - val_accu
racy: 0.7083
```

```
Epoch 33/50
27/27 [==============================] - 128s 5s/step - loss: 0.1026 - accuracy: 0.9745 - val_loss: 1.9420 - val_accu
racy: 0.7407
Epoch 34/50
27/27 [==============================] - 128s 5s/step - loss: 0.0869 - accuracy: 0.9734 - val_loss: 2.4203 - val_accu
racy: 0.7083
Epoch 35/50
27/27 [==============================] - 129s 5s/step - loss: 0.0603 - accuracy: 0.9861 - val_loss: 3.0703 - val_accu
racy: 0.7037
Epoch 36/50
27/27 [==============================] - 128s 5s/step - loss: 0.0714 - accuracy: 0.9826 - val_loss: 2.5122 - val_accu
racy: 0.6944
Epoch 37/50
27/27 [==============================] - 129s 5s/step - loss: 0.0567 - accuracy: 0.9861 - val_loss: 1.8895 - val_accu
racy: 0.7176
Epoch 38/50
27/27 [==============================] - 129s 5s/step - loss: 0.0453 - accuracy: 0.9850 - val_loss: 3.3969 - val_accu
racy: 0.6852
Epoch 39/50
27/27 [==============================] - 128s 5s/step - loss: 0.0663 - accuracy: 0.9699 - val_loss: 6.9810 - val_accu
racy: 0.6806
Epoch 40/50
27/27 [==============================] - 128s 5s/step - loss: 0.0321 - accuracy: 0.9954 - val_loss: 4.7027 - val_accu
racy: 0.7083
Epoch 41/50
27/27 [==============================] - 128s 5s/step - loss: 0.0847 - accuracy: 0.9815 - val_loss: 7.6695 - val_accu
racy: 0.7037
Epoch 42/50
27/27 [==============================] - 128s 5s/step - loss: 0.1251 - accuracy: 0.9699 - val_loss: 4.5016 - val_accu
racy: 0.6806
Epoch 43/50
27/27 [==============================] - 129s 5s/step - loss: 0.0256 - accuracy: 0.9931 - val_loss: 6.5038 - val_accu
racy: 0.7130
Epoch 44/50
27/27 [==============================] - 129s 5s/step - loss: 0.0819 - accuracy: 0.9861 - val_loss: 7.0234 - val_accu
racy: 0.6435
Epoch 45/50
27/27 [==============================] - 129s 5s/step - loss: 0.0861 - accuracy: 0.9792 - val_loss: 4.9212 - val_accu
racy: 0.7361
Epoch 46/50
27/27 [==============================] - 129s 5s/step - loss: 0.0210 - accuracy: 0.9919 - val_loss: 2.6663 - val_accu
racy: 0.7361
Epoch 47/50
27/27 [==============================] - 128s 5s/step - loss: 0.0975 - accuracy: 0.9734 - val_loss: 3.0830 - val_accu
racy: 0.7454
Epoch 48/50
27/27 [==============================] - 128s 5s/step - loss: 0.1557 - accuracy: 0.9792 - val_loss: 3.7367 - val_accu
racy: 0.6806
Epoch 49/50
27/27 [==============================] - 128s 5s/step - loss: 0.0305 - accuracy: 0.9873 - val_loss: 3.8291 - val_accu
racy: 0.6759
Epoch 50/50
27/27 [==============================] - 128s 5s/step - loss: 0.1048 - accuracy: 0.9803 - val_loss: 5.2759 - val_accu
racy: 0.5926
----------------Test accuracy for 4 fold----------------
Confusion Matrix :
 [[23  0  0  7]
 [ 6 17  0  7]
 [ 7  0  5 18]
 [ 6  0  0 24]]
Accuracy    :  0.575
Specificity :  0.823026973026973
Sensitivity :  0.5750000000000001
----------------------------End of 4 Fold--------------------------
-----------------------Start of 5 Fold-----------------------
test images for normal class from 120 150
test images for cataract class from 120 150
test images for glaucoma class from 120 150
test images for retina disease class from 120 150
train_validation images for normal class from 0 to 120 and 150 to 300
train_validation images for cataract class from 0 to 120 and 150 to 300
train_validation  images for glaucoma class from 0 120 and 150 to 300
train_validation images for retina disease class from 0 to 120 and 150 to 300
model building and compiling for fold 5
Epoch 1/50
27/27 [==============================] - 143s 5s/step - loss: 13.7020 - accuracy: 0.4294 - val_loss: 67.3890 - val_ac
curacy: 0.3657
Epoch 2/50
27/27 [==============================] - 126s 5s/step - loss: 5.2120 - accuracy: 0.5544 - val_loss: 16.1572 - val_acc
uracy: 0.4491
Epoch 3/50
27/27 [==============================] - 127s 5s/step - loss: 3.8196 - accuracy: 0.5498 - val_loss: 21.9281 - val_acc
uracy: 0.3333
Epoch 4/50
27/27 [==============================] - 128s 5s/step - loss: 2.5604 - accuracy: 0.6111 - val_loss: 1.9103 - val_accu
racy: 0.5417
Epoch 5/50
27/27 [==============================] - 127s 5s/step - loss: 1.4478 - accuracy: 0.6678 - val_loss: 2.3883 - val_accu
```

```
racy: 0.5231
Epoch 6/50
27/27 [==============================] - 128s 5s/step - loss: 1.0895 - accuracy: 0.7477 - val_loss: 2.9046 - val_accu
racy: 0.5648
Epoch 7/50
27/27 [==============================] - 131s 5s/step - loss: 1.1761 - accuracy: 0.7836 - val_loss: 3.4931 - val_accu
racy: 0.5648
Epoch 8/50
27/27 [==============================] - 136s 5s/step - loss: 0.9202 - accuracy: 0.8090 - val_loss: 2.4550 - val_accu
racy: 0.6157
Epoch 9/50
27/27 [==============================] - 127s 5s/step - loss: 0.5770 - accuracy: 0.8472 - val_loss: 4.5846 - val_accu
racy: 0.5370
Epoch 10/50
27/27 [==============================] - 134s 5s/step - loss: 0.4176 - accuracy: 0.8727 - val_loss: 2.5676 - val_accu
racy: 0.5278
Epoch 11/50
27/27 [==============================] - 129s 5s/step - loss: 0.3211 - accuracy: 0.9097 - val_loss: 4.0960 - val_accu
racy: 0.5093
Epoch 12/50
27/27 [==============================] - 129s 5s/step - loss: 0.2500 - accuracy: 0.9201 - val_loss: 4.1966 - val_accu
racy: 0.5046
Epoch 13/50
27/27 [==============================] - 118s 4s/step - loss: 0.3573 - accuracy: 0.9016 - val_loss: 1.8798 - val_accu
racy: 0.6898
Epoch 14/50
27/27 [==============================] - 125s 5s/step - loss: 0.1834 - accuracy: 0.9560 - val_loss: 2.1403 - val_accu
racy: 0.6620
Epoch 15/50
27/27 [==============================] - 124s 5s/step - loss: 0.2359 - accuracy: 0.9306 - val_loss: 2.5124 - val_accu
racy: 0.6528
Epoch 16/50
27/27 [==============================] - 130s 5s/step - loss: 0.2575 - accuracy: 0.9421 - val_loss: 3.2979 - val_accu
racy: 0.5972
Epoch 17/50
27/27 [==============================] - 125s 5s/step - loss: 0.3214 - accuracy: 0.9201 - val_loss: 2.1105 - val_accu
racy: 0.6343
Epoch 18/50
27/27 [==============================] - 131s 5s/step - loss: 0.1795 - accuracy: 0.9549 - val_loss: 3.3649 - val_accu
racy: 0.6204
Epoch 19/50
27/27 [==============================] - 136s 5s/step - loss: 0.1179 - accuracy: 0.9595 - val_loss: 2.1855 - val_accu
racy: 0.6389
Epoch 20/50
27/27 [==============================] - 129s 5s/step - loss: 0.1148 - accuracy: 0.9595 - val_loss: 2.1078 - val_accu
racy: 0.7083
Epoch 21/50
27/27 [==============================] - 129s 5s/step - loss: 0.0890 - accuracy: 0.9688 - val_loss: 3.0969 - val_accu
racy: 0.6759
Epoch 22/50
27/27 [==============================] - 128s 5s/step - loss: 0.1261 - accuracy: 0.9688 - val_loss: 2.6265 - val_accu
racy: 0.6759
Epoch 23/50
27/27 [==============================] - 139s 5s/step - loss: 0.0662 - accuracy: 0.9803 - val_loss: 2.2558 - val_accu
racy: 0.6944
Epoch 24/50
27/27 [==============================] - 133s 5s/step - loss: 0.1105 - accuracy: 0.9745 - val_loss: 2.4074 - val_accu
racy: 0.7083
Epoch 25/50
27/27 [==============================] - 126s 5s/step - loss: 0.0922 - accuracy: 0.9780 - val_loss: 1.6977 - val_accu
racy: 0.7269
Epoch 26/50
27/27 [==============================] - 129s 5s/step - loss: 0.1050 - accuracy: 0.9722 - val_loss: 2.6464 - val_accu
racy: 0.6528
Epoch 27/50
27/27 [==============================] - 129s 5s/step - loss: 0.0491 - accuracy: 0.9826 - val_loss: 4.5417 - val_accu
racy: 0.6759
Epoch 28/50
27/27 [==============================] - 129s 5s/step - loss: 0.0754 - accuracy: 0.9745 - val_loss: 3.1871 - val_accu
racy: 0.6806
Epoch 29/50
27/27 [==============================] - 130s 5s/step - loss: 0.1325 - accuracy: 0.9722 - val_loss: 2.2373 - val_accu
racy: 0.6852
Epoch 30/50
27/27 [==============================] - 129s 5s/step - loss: 0.0725 - accuracy: 0.9838 - val_loss: 2.6513 - val_accu
racy: 0.6898
Epoch 31/50
27/27 [==============================] - 129s 5s/step - loss: 0.0554 - accuracy: 0.9815 - val_loss: 3.7220 - val_accu
racy: 0.6667
Epoch 32/50
27/27 [==============================] - 128s 5s/step - loss: 0.1160 - accuracy: 0.9618 - val_loss: 2.3612 - val_accu
racy: 0.7269
Epoch 33/50
27/27 [==============================] - 129s 5s/step - loss: 0.0517 - accuracy: 0.9826 - val_loss: 1.8230 - val_accu
racy: 0.7222
Epoch 34/50
27/27 [==============================] - 122s 5s/step - loss: 0.0492 - accuracy: 0.9838 - val_loss: 3.0471 - val_accu
racy: 0.6759
```

```
Epoch 35/50
27/27 [==============================] - 128s 5s/step - loss: 0.0586 - accuracy: 0.9873 - val_loss: 2.6858 - val_accu
racy: 0.7361
Epoch 36/50
27/27 [==============================] - 128s 5s/step - loss: 0.0674 - accuracy: 0.9850 - val_loss: 2.4149 - val_accu
racy: 0.6991
Epoch 37/50
27/27 [==============================] - 128s 5s/step - loss: 0.0438 - accuracy: 0.9873 - val_loss: 2.1280 - val_accu
racy: 0.6620
Epoch 38/50
27/27 [==============================] - 128s 5s/step - loss: 0.0757 - accuracy: 0.9826 - val_loss: 2.2775 - val_accu
racy: 0.6713
Epoch 39/50
27/27 [==============================] - 125s 5s/step - loss: 0.0491 - accuracy: 0.9826 - val_loss: 3.0572 - val_accu
racy: 0.6667
Epoch 40/50
27/27 [==============================] - 125s 5s/step - loss: 0.0379 - accuracy: 0.9896 - val_loss: 3.2794 - val_accu
racy: 0.6574
Epoch 41/50
27/27 [==============================] - 122s 5s/step - loss: 0.0311 - accuracy: 0.9919 - val_loss: 4.2183 - val_accu
racy: 0.6343
Epoch 42/50
27/27 [==============================] - 120s 4s/step - loss: 0.0557 - accuracy: 0.9850 - val_loss: 2.0361 - val_accu
racy: 0.7454
Epoch 43/50
27/27 [==============================] - 125s 5s/step - loss: 0.0446 - accuracy: 0.9873 - val_loss: 2.5411 - val_accu
racy: 0.7407
Epoch 44/50
27/27 [==============================] - 121s 4s/step - loss: 0.0461 - accuracy: 0.9861 - val_loss: 2.5111 - val_accu
racy: 0.6991
Epoch 45/50
27/27 [==============================] - 117s 4s/step - loss: 0.0494 - accuracy: 0.9873 - val_loss: 2.4577 - val_accu
racy: 0.6991
Epoch 46/50
27/27 [==============================] - 119s 4s/step - loss: 0.0754 - accuracy: 0.9792 - val_loss: 2.7911 - val_accu
racy: 0.7037
Epoch 47/50
27/27 [==============================] - 121s 5s/step - loss: 0.1093 - accuracy: 0.9664 - val_loss: 2.2225 - val_accu
racy: 0.7269
Epoch 48/50
27/27 [==============================] - 116s 4s/step - loss: 0.0353 - accuracy: 0.9896 - val_loss: 4.3709 - val_accu
racy: 0.6296
Epoch 49/50
27/27 [==============================] - 119s 4s/step - loss: 0.0725 - accuracy: 0.9815 - val_loss: 4.3048 - val_accu
racy: 0.6157
Epoch 50/50
27/27 [==============================] - 122s 5s/step - loss: 0.0427 - accuracy: 0.9896 - val_loss: 2.3026 - val_accu
racy: 0.7176
----------------Test accuracy for 5 fold----------------
Confusion Matrix :
 [[22  3  5  0]
 [ 0 24  2  4]
 [ 2  1 20  7]
 [ 7  2  5 16]]
Accuracy    :  0.6833333333333333
Specificity :  0.8676989780929998
Sensitivity :  0.6833333333333332
-----------------------------End of 5 Fold-------------------------
```

Test Evaluation Results

In [17]:
```
test_accuracy
```

Out[17]:
```
[0.625, 0.675, 0.5666666666666667, 0.575, 0.6833333333333333]
```

In [18]:
```
mean_test_accuracy=np.mean(test_accuracy)
mean_test_accuracy
```

Out[18]:
```
0.625
```

In [19]:
```
test_sensitivity
```

Out[19]:
```
[0.625,
 0.6749999999999999,
 0.5666666666666667,
 0.5750000000000001,
 0.6833333333333332]
```

In [20]:
```
mean_test_sensitivity= np.mean(test_sensitivity)
mean_test_sensitivity
```

Out[20]:
```
0.6249999999999999
```

In [21]:
```
test_specificity
```

Out[21]:
```
[0.8400558523555124,
 0.8653964086368978,
 0.8127085800994504,
 0.823026973026973,
 0.8676989780929998]
```

In [22]:
```
mean_test_specificity= np.mean(test_specificity)
mean_test_specificity
```

Out[22]:
```
0.8417773584423667
```

Training and Validation Evaluation Results

In [23]:
```
train_acc
```

Out[23]:
```
array([0.42824075, 0.5300926 , 0.60185188, 0.63078701, 0.71064812,
       0.71412039, 0.77430558, 0.82291669, 0.80092591, 0.88657409,
       0.90625   , 0.93981481, 0.90856481, 0.9386574 , 0.94212961,
       0.9386574 , 0.96064812, 0.94328701, 0.9699074 , 0.97222221,
       0.96643519, 0.96296299, 0.96527779, 0.96296299, 0.9699074 ,
       0.9548611 , 0.9826389 , 0.97685188, 0.99074072, 0.96875   ,
       0.9699074 , 0.97453701, 0.9826389 , 0.97106481, 0.9849537 ,
       0.9826389 , 0.97685188, 0.98032409, 0.9849537 , 0.98842591,
       0.9849537 , 0.98726851, 0.97800928, 0.96875   , 0.9837963 ,
       0.9849537 , 0.99189812, 0.9849537 , 0.97222221, 0.98726851,
       0.42592594, 0.54976851, 0.5949074 , 0.62268519, 0.70601851,
       0.75694442, 0.80555558, 0.85185188, 0.85185188, 0.86574072,
       0.91898149, 0.9050926 , 0.9375    , 0.89467591, 0.93981481,
       0.94675928, 0.93402779, 0.95717591, 0.97106481, 0.92592591,
       0.96296299, 0.98148149, 0.97685188, 0.97569442, 0.9675926 ,
       0.94907409, 0.95717591, 0.97222221, 0.96643519, 0.97685188,
       0.97916669, 0.98032409, 0.95833331, 0.98032409, 0.98842591,
       0.97222221, 0.9849537 , 0.98958331, 0.9675926 , 0.9837963 ,
       0.98958331, 0.97916669, 0.98726851, 0.98842591, 0.9861111 ,
       0.98958331, 0.98726851, 0.98726851, 0.99305558, 0.97916669,
       0.44328704, 0.52314812, 0.57060188, 0.59953701, 0.67592591,
       0.75      , 0.72337961, 0.7986111 , 0.83564812, 0.8576389 ,
       0.86689812, 0.90046299, 0.91203701, 0.9224537 , 0.91203701,
       0.95023149, 0.93287039, 0.9525463 , 0.94097221, 0.9699074 ,
       0.9699074 , 0.95717591, 0.94907409, 0.97685188, 0.96875   ,
       0.95833331, 0.97106481, 0.97337961, 0.9675926 , 0.97106481,
       0.97106481, 0.97685188, 0.97569442, 0.99421299, 0.97106481,
       0.97106481, 0.97916669, 0.97685188, 0.98032409, 0.99189812,
       0.98148149, 0.97685188, 0.97569442, 0.97685188, 0.98842591,
       0.98958331, 0.98842591, 0.98726851, 0.99189812, 0.97569442,
       0.4212963 , 0.53356481, 0.5625    , 0.62152779, 0.70601851,
       0.71064812, 0.75347221, 0.84143519, 0.81828701, 0.86574072,
       0.8761574 , 0.90162039, 0.9074074 , 0.94328701, 0.92592591,
       0.94212961, 0.94560188, 0.95601851, 0.95023149, 0.96412039,
       0.98148149, 0.96643519, 0.96527779, 0.97337961, 0.96527779,
       0.97453701, 0.97569442, 0.97800928, 0.97106481, 0.97222221,
       0.98032409, 0.97453701, 0.97453701, 0.97337961, 0.9861111 ,
       0.9826389 , 0.9861111 , 0.9849537 , 0.9699074 , 0.99537039,
       0.98148149, 0.9699074 , 0.99305558, 0.9861111 , 0.97916669,
       0.99189812, 0.97337961, 0.97916669, 0.98726851, 0.98032409,
       0.42939815, 0.55439812, 0.54976851, 0.6111111 , 0.66782409,
       0.74768519, 0.78356481, 0.80902779, 0.84722221, 0.87268519,
       0.90972221, 0.9201389 , 0.90162039, 0.95601851, 0.93055558,
       0.94212961, 0.9201389 , 0.9548611 , 0.95949072, 0.95949072,
       0.96875   , 0.96875   , 0.98032409, 0.97453701, 0.97800928,
       0.97222221, 0.9826389 , 0.97453701, 0.97222221, 0.9837963 ,
       0.98148149, 0.96180558, 0.9826389 , 0.9837963 , 0.98726851,
       0.9849537 , 0.98726851, 0.9826389 , 0.9826389 , 0.98958331,
       0.99189812, 0.9849537 , 0.98726851, 0.9861111 , 0.98726851,
       0.97916669, 0.96643519, 0.98958331, 0.98148149, 0.98958331])
```

In [24]:
```
mean_train_accuracy=np.mean(train_acc)
mean_train_accuracy
```

Out[24]:
```
0.9112592575550079
```

In [25]:
```
val_acc
```

Out[25]:
```
array([0.45833334, 0.48148149, 0.46296296, 0.46296296, 0.44907406,
       0.4861111 , 0.40277779, 0.55092591, 0.5138889 , 0.43981481,
       0.5       , 0.52777779, 0.75925928, 0.52777779, 0.62037039,
       0.71296299, 0.62962961, 0.6111111 , 0.58796299, 0.5925926 ,
       0.54166669, 0.62037039, 0.62037039, 0.69907409, 0.68518519,
       0.64814812, 0.69444442, 0.66203701, 0.72222221, 0.71296299,
       0.65277779, 0.56481481, 0.66203701, 0.6388889 , 0.69907409,
       0.60648149, 0.71296299, 0.71296299, 0.71296299, 0.72222221,
       0.71296299, 0.69907409, 0.5925926 , 0.6388889 , 0.69444442,
       0.68981481, 0.69907409, 0.68055558, 0.70833331, 0.68981481,
       0.46296296, 0.50925928, 0.60648149, 0.4537037 , 0.49537036,
       0.54166669, 0.49074075, 0.55092591, 0.6111111 , 0.53703701,
       0.43518519, 0.56018519, 0.49074075, 0.62962961, 0.6712963 ,
       0.7037037 , 0.69907409, 0.61574072, 0.60648149, 0.66203701,
       0.76851851, 0.70833331, 0.7175926 , 0.68981481, 0.60648149,
       0.66203701, 0.70833331, 0.77314812, 0.72222221, 0.65277779,
       0.69907409, 0.56018519, 0.5462963 , 0.69907409, 0.70833331,
       0.69907409, 0.77314812, 0.7361111 , 0.72222221, 0.72685188,
       0.74074072, 0.74074072, 0.56018519, 0.64351851, 0.7037037 ,
       0.72222221, 0.7175926 , 0.69444442, 0.72685188, 0.75925928,
       0.43518519, 0.53703701, 0.55092591, 0.44907406, 0.55555558,
       0.46296296, 0.34722221, 0.51851851, 0.49537036, 0.5       ,
       0.6111111 , 0.58333331, 0.71296299, 0.60648149, 0.4675926 ,
       0.69444442, 0.67592591, 0.64814812, 0.69444442, 0.64351851,
       0.625     , 0.66666669, 0.66666669, 0.62962961, 0.68055558,
       0.6574074 , 0.62962961, 0.6388889 , 0.66666669, 0.63425928,
       0.6574074 , 0.73148149, 0.66203701, 0.68981481, 0.69444442,
       0.62962961, 0.69444442, 0.6388889 , 0.7037037 , 0.68981481,
       0.69444442, 0.60185188, 0.67592591, 0.6712963 , 0.64814812,
       0.68981481, 0.68981481, 0.61574072, 0.69444442, 0.6712963 ,
       0.49074075, 0.4212963 , 0.63425928, 0.52777779, 0.4537037 ,
       0.39814815, 0.45833334, 0.55555558, 0.4212963 , 0.625     ,
       0.60185188, 0.5462963 , 0.68518519, 0.7361111 , 0.625     ,
       0.56018519, 0.64351851, 0.65277779, 0.73148149, 0.69444442,
       0.68055558, 0.68518519, 0.6574074 , 0.7037037 , 0.6712963 ,
       0.70833331, 0.65277779, 0.72685188, 0.7175926 , 0.69444442,
       0.71296299, 0.70833331, 0.74074072, 0.70833331, 0.7037037 ,
       0.69444442, 0.7175926 , 0.68518519, 0.68055558, 0.70833331,
       0.7037037 , 0.68055558, 0.71296299, 0.64351851, 0.7361111 ,
       0.7361111 , 0.74537039, 0.68055558, 0.67592591, 0.5925926 ,
       0.36574075, 0.44907406, 0.33333334, 0.54166669, 0.52314812,
       0.56481481, 0.56481481, 0.61574072, 0.53703701, 0.52777779,
       0.50925928, 0.50462961, 0.68981481, 0.66203701, 0.65277779,
       0.59722221, 0.63425928, 0.62037039, 0.6388889 , 0.70833331,
       0.67592591, 0.67592591, 0.69444442, 0.70833331, 0.72685188,
       0.65277779, 0.67592591, 0.68055558, 0.68518519, 0.68981481,
       0.66666669, 0.72685188, 0.72222221, 0.67592591, 0.7361111 ,
       0.69907409, 0.66203701, 0.6712963 , 0.66666669, 0.6574074 ,
       0.63425928, 0.74537039, 0.74074072, 0.69907409, 0.69907409,
       0.7037037 , 0.72685188, 0.62962961, 0.61574072, 0.7175926 ])
```

In [26]:
```
mean_val_accuracy=np.mean(val_acc)
mean_val_accuracy
```

Out[26]:
```
0.6338333328962326
```

In [27]:
```
train_loss
```

Out[27]:
```
array([14.01601601,  5.99046373,  3.46708083,  1.88818371,  1.48318422,
        1.53338933,  1.07926536,  0.84145361,  0.82222039,  0.51442599,
        0.39413667,  0.18971464,  0.33293816,  0.20418647,  0.17163716,
        0.20579287,  0.17036383,  0.25432554,  0.11076803,  0.1539181 ,
        0.09643257,  0.16374858,  0.14883578,  0.17089194,  0.14235191,
        0.18202019,  0.07156228,  0.09606129,  0.03795874,  0.14063142,
        0.1158237 ,  0.08651969,  0.14393376,  0.12074108,  0.04751857,
        0.05595391,  0.31473139,  0.08472449,  0.0558994 ,  0.03433263,
        0.0713957 ,  0.04540246,  0.12224007,  0.28854302,  0.17643224,
        0.08721486,  0.04682158,  0.05839019,  0.13016254,  0.09579504,
       12.75337601,  4.21546221,  2.2095015 ,  2.13000774,  1.56619751,
        1.10082507,  1.06542432,  0.792247  ,  0.7477901 ,  0.54673469,
        0.41347012,  0.39398646,  0.21596012,  0.60878134,  0.28041005,
        0.22389957,  0.24804316,  0.13916098,  0.17246431,  0.28095976,
        0.13614315,  0.24987176,  0.0937829 ,  0.09870087,  0.3805773 ,
        0.15390912,  0.34057918,  0.14000872,  0.1750287 ,  0.18645746,
        0.08375872,  0.10117275,  0.24058726,  0.08095474,  0.03855216,
        0.09859791,  0.07517339,  0.05227556,  0.14479533,  0.06207924,
        0.03242755,  0.17634432,  0.07449857,  0.03973809,  0.06187151,
        0.03741236,  0.04488313,  0.05626541,  0.05321465,  0.11538363,
       13.89763641,  5.02031517,  4.65163803,  2.03166056,  1.6488874 ,
        1.36429656,  1.60345888,  1.16728175,  0.63168865,  0.74211937,
        0.64894539,  0.42063585,  0.54614741,  0.37693322,  0.50586748,
        0.19672887,  0.2670995 ,  0.16046506,  0.27227071,  0.19389051,
        0.12759095,  0.16846628,  0.22161125,  0.09475384,  0.14979258,
        0.20607138,  0.16476227,  0.10788175,  0.0987441 ,  0.12738079,
        0.08915047,  0.08009963,  0.11108717,  0.03111496,  0.1549824 ,
        0.10419611,  0.06121275,  0.07316733,  0.0674139 ,  0.03081711,
        0.09532586,  0.11196449,  0.06365303,  0.11490574,  0.02929034,
        0.03541635,  0.0502861 ,  0.04952049,  0.03930816,  0.11762076,
       13.82173538,  5.59660292,  3.33028507,  2.51855278,  2.00264955,
        1.43038774,  1.21505058,  0.72488928,  0.75382948,  0.54726225,
        0.62343431,  0.43467048,  0.33925977,  0.21991007,  0.29755104,
        0.21629547,  0.2326546 ,  0.16765939,  0.16094697,  0.15712602,
        0.05278164,  0.11700945,  0.14188373,  0.08614429,  0.08677029,
        0.10593868,  0.11390372,  0.08160556,  0.14464806,  0.11939321,
        0.06648253,  0.11800092,  0.1025809 ,  0.0868568 ,  0.0602524 ,
        0.07144428,  0.056683  ,  0.0453496 ,  0.0663213 ,  0.03206278,
        0.08472499,  0.12508985,  0.02561633,  0.08187032,  0.08613941,
        0.02098496,  0.0975494 ,  0.15571481,  0.03049796,  0.10475449,
       13.70202637,  5.21201468,  3.81959319,  2.56044197,  1.44783032,
        1.08954442,  1.17613208,  0.92018098,  0.57697314,  0.41764462,
        0.32109949,  0.25000983,  0.35734209,  0.18343326,  0.23594482,
        0.25750372,  0.32136261,  0.17953585,  0.11788411,  0.11483736,
        0.08896166,  0.12610556,  0.06621944,  0.11046496,  0.09216829,
        0.10500902,  0.04905976,  0.07541762,  0.13252272,  0.07250758,
        0.05538414,  0.11599738,  0.05173727,  0.04919526,  0.05859042,
        0.06738885,  0.04382779,  0.07570911,  0.04912584,  0.03793006,
        0.0311033 ,  0.05573113,  0.04455024,  0.04613318,  0.04938873,
        0.07543512,  0.10927243,  0.03530414,  0.0724903 ,  0.04271918])
```

In [28]:
```
mean_train_loss=np.mean(train_loss)
mean_train_loss
```

Out[28]:
```
0.7320737297609449
```

In [29]:
```
val_loss
```

Out[29]:
```
array([ 35.82098007,   6.76688528,   3.36643195,   4.16696835,
         2.86958027,   2.39051986,   3.23010707,   3.08033347,
         4.76453829,   6.7334919 ,   6.06546545,   4.36743116,
         1.37073815,   3.38882256,   2.68051553,   2.36681008,
         2.54374528,   3.04979348,   3.20193815,   4.13996744,
         4.15596104,   3.72309756,   3.81311035,   3.58289623,
         2.99094987,   4.69102669,   2.93203449,   3.19594073,
         3.01362133,   3.50500059,   5.09752464, 105.40859222,
         3.8572104 ,   5.72795963,  13.49778938,  79.6286087 ,
         3.55446959,   3.24195838,   3.91282988,   3.64913082,
         3.03464222,   3.22578335,  28.51343536,   4.89647198,
         4.85424519,   3.6787703 ,   3.59648132,   3.70778251,
         3.40488315,   4.5308814 ,  20.98736954,   6.2493906 ,
         1.88639557,   3.52640343,   2.74864221,   1.98784018,
         5.10121965,   2.93294168,   2.70755959,   4.83800602,
         5.53572607,   4.44125319,   5.43979168,   2.96912408,
         2.78345633,   3.28101301,   2.66042161,   3.35749459,
         4.76512957,   3.53011727,   2.45221114,   4.29099178,
         3.42474651,   3.63989162,   5.70350742,   4.20752621,
         3.25804424,   2.65622234,   3.20052814,   3.95159554,
         3.85728741,   5.11065435,   6.39353037,   3.62923479,
         4.7817235 ,   3.56325364,   2.45935082,   3.54749703,
         3.30039477,   3.07463932,   2.92457628,   2.70416832,
         9.0966568 ,   7.8802557 ,   7.83487511,   4.82302809,
         4.22216558,   4.04813242,   5.57975531,   3.34637737,
        35.62550354,   9.31854343,   6.28504896,   3.48928499,
         2.64398861,   6.25956202,   6.85523987,   4.18135738,
         3.7405026 ,   3.16674089,   2.67481875,   3.42327762,
         1.88463533,   2.88975668,   8.55189896,   2.65250897,
         3.82999325,   3.42729855,   2.35621762,   3.42101169,
         3.39677382,   3.2326498 ,   3.61131263,   3.46896863,
         2.90707684,   4.24974012,   5.71885157,   4.33051634,
         4.45122576,   5.30963516,   4.13691998,   2.62708545,
         4.25935745,   4.31961632,   2.98148298,   5.41097355,
         3.41536379,   4.98991013,   3.8289516 ,   4.30832624,
         3.48008871,   5.07438612,   4.70381451,   4.16675997,
         5.88802433,   5.0817914 ,   7.56911325,   6.296978  ,
         3.23272491,   3.89830565,  35.26340103,   8.52641392,
         3.06659198,   2.1874826 ,   3.77316284,   5.94704199,
         5.8028326 ,   3.30478477,   5.68832016,   2.75733209,
         2.3623662 ,   2.14866972,   1.68206763,   1.50243211,
         3.18562603,   3.02221513,   3.13095999,   2.53010011,
         2.3285892 ,   2.04899883,   2.21393514,   2.48871565,
         2.18511033,   2.6632967 ,   2.4715209 ,   2.60612178,
         2.49388146,   2.18072701,   2.82012773,   2.78612947,
         2.56955528,   2.14802885,   1.941998  ,   2.42030621,
         3.07032037,   2.51216555,   1.88952398,   3.39693546,
         6.98104239,   4.70272636,   7.66947746,   4.50163746,
         6.50381851,   7.02339077,   4.92122889,   2.66630101,
         3.08295584,   3.73672509,   3.82906699,   5.27591848,
        67.38898468,  16.15716934,  21.92812729,   1.91027355,
         2.38827252,   2.90463257,   3.49314523,   2.45499825,
         4.58463192,   2.56755114,   4.09604836,   4.19661474,
         1.8798213 ,   2.14033484,   2.51239014,   3.29793262,
         2.11046028,   3.36487007,   2.18547416,   2.10784245,
         3.09691811,   2.62646532,   2.25582933,   2.40741277,
         1.69773161,   2.64644027,   4.54169083,   3.1870544 ,
         2.23734903,   2.65130711,   3.72203684,   2.36123872,
         1.82302022,   3.04708886,   2.68583226,   2.41490841,
         2.12799168,   2.27747536,   3.05724549,   3.27940702,
         4.21833181,   2.03610778,   2.54109859,   2.51111197,
         2.45765424,   2.79112434,   2.22248411,   4.37091398,
         4.30482388,   2.30258226])
```

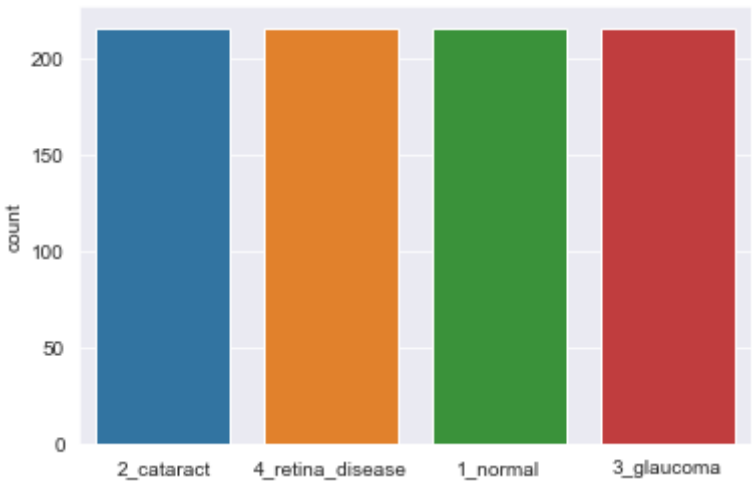
In [30]:
```
mean_val_loss=np.mean(val_loss)
mean_val_loss
```

Out[30]:
```
5.377146062850952
```

Plot to Visualize the Number of Images in Each Label of Trainig Dataset

In [31]:
```
l = []
for i in train:
    if(i[1] == 0):
        l.append("1_normal")

    elif (i[1] == 1):
        l.append("2_cataract")

    elif (i[1] == 2):
        l.append("3_glaucoma")

    else :
        l.append("4_retina_disease")


sns.set_style('darkgrid')
sns.countplot(l)
```

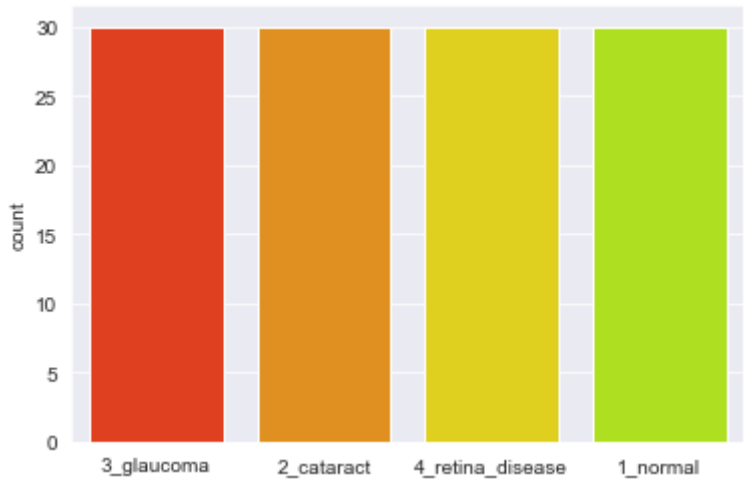
Out[31]: <matplotlib.axes._subplots.AxesSubplot at 0x2a50000b8b0>



Plot to Visualize the Number of Images in Each Label of Test Dataset.

In [32]:
```
l = []
for i in test:
    if(i[1] == 0):
        l.append("1_normal")

    elif (i[1] == 1):
        l.append("2_cataract")

    elif (i[1] == 2):
        l.append("3_glaucoma")

    else :
        l.append("4_retina_disease")


sns.set_style('darkgrid')
sns.countplot(l,palette='prism')
```

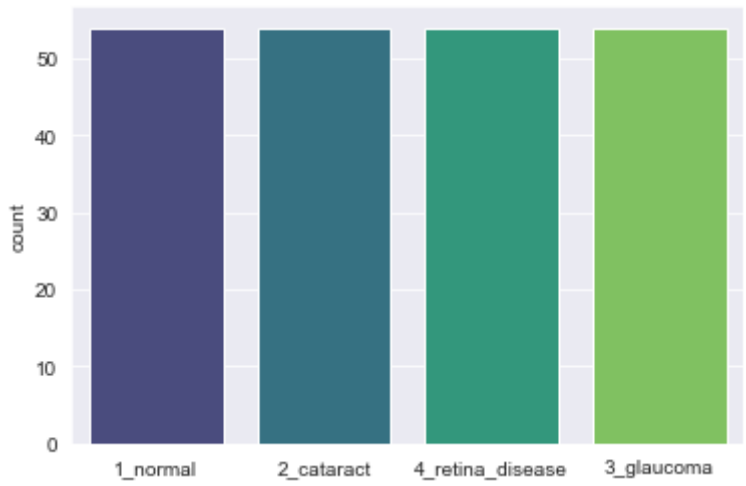
Out[32]: <matplotlib.axes._subplots.AxesSubplot at 0x2a5249e7670>



Plot to Visualize the Number of Images in Each Label of Validation Dataset.

In [33]:
```
l = []
for i in validation:
    if(i[1] == 0):
        l.append("1_normal")

    elif (i[1] == 1):
        l.append("2_cataract")

    elif (i[1] == 2):
        l.append("3_glaucoma")

    else :
        l.append("4_retina_disease")


sns.set_style('darkgrid')
sns.countplot(l,palette='viridis')
```

Out[33]: <matplotlib.axes._subplots.AxesSubplot at 0x2a50ebaa160>

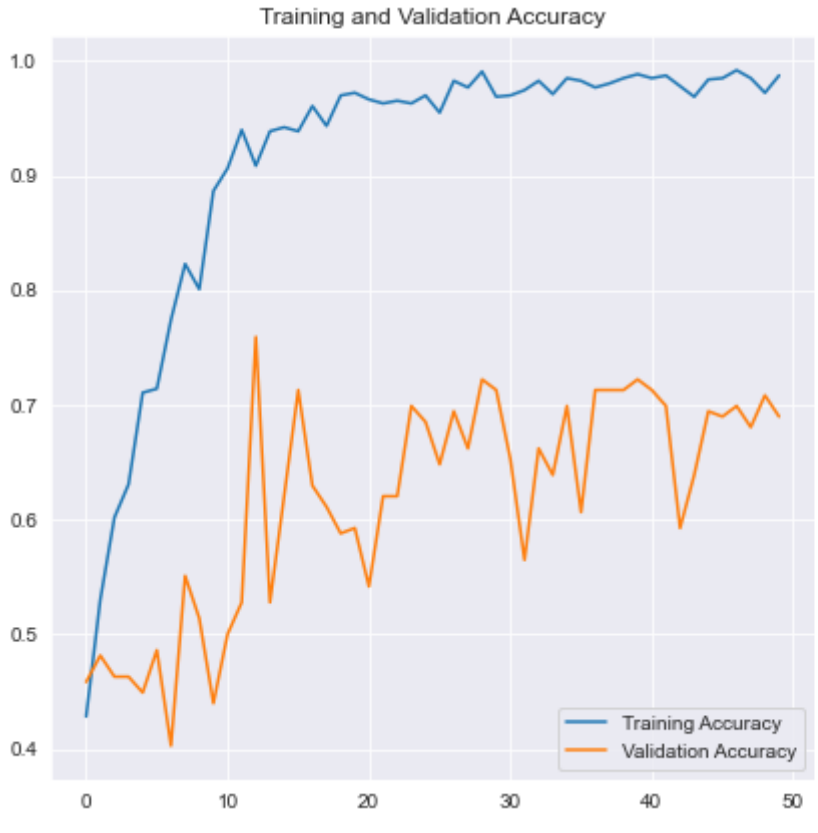


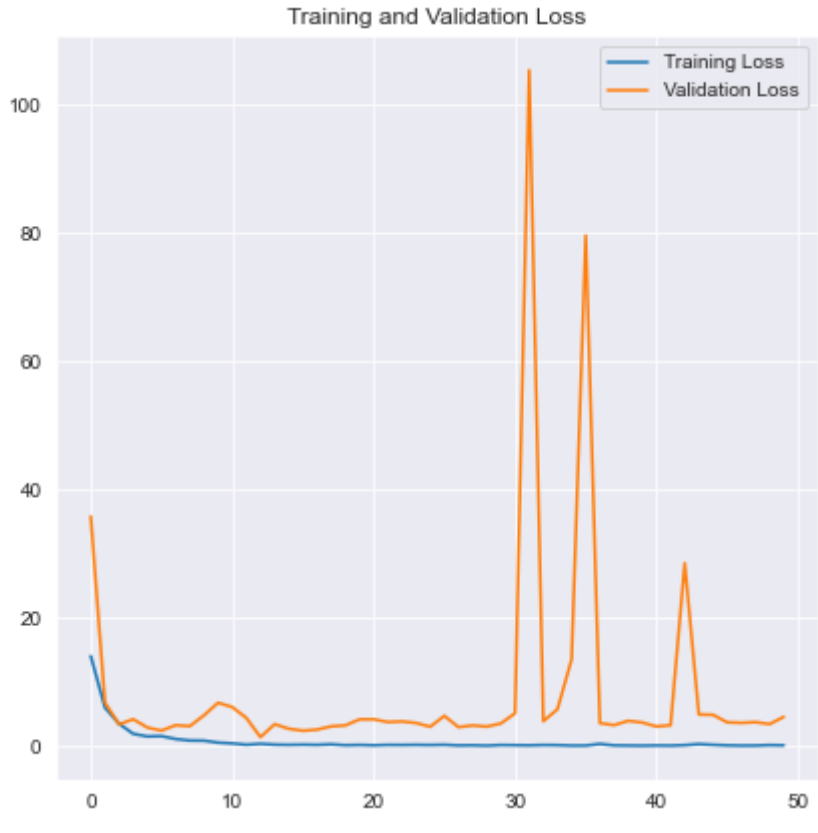
Training,Validation Accuracy and Loss Plot for 50 Epochs

In [34]:
```
def plot_print(i,j):
    epochs_range = range(50)

    plt.figure(figsize=(15, 15))
    plt.subplot(2, 2, 1)
    plt.plot(epochs_range, train_acc[i:j], label='Training Accuracy')
    plt.plot(epochs_range, val_acc[i:j], label='Validation Accuracy')
    plt.legend(loc='lower right')
    plt.title('Training and Validation Accuracy')

    plt.subplot(2, 2, 2)
    plt.plot(epochs_range, train_loss[i:j], label='Training Loss')
    plt.plot(epochs_range, val_loss[i:j], label='Validation Loss')
    plt.legend(loc='upper right')
    plt.title('Training and Validation Loss')

    return plt.show()
```

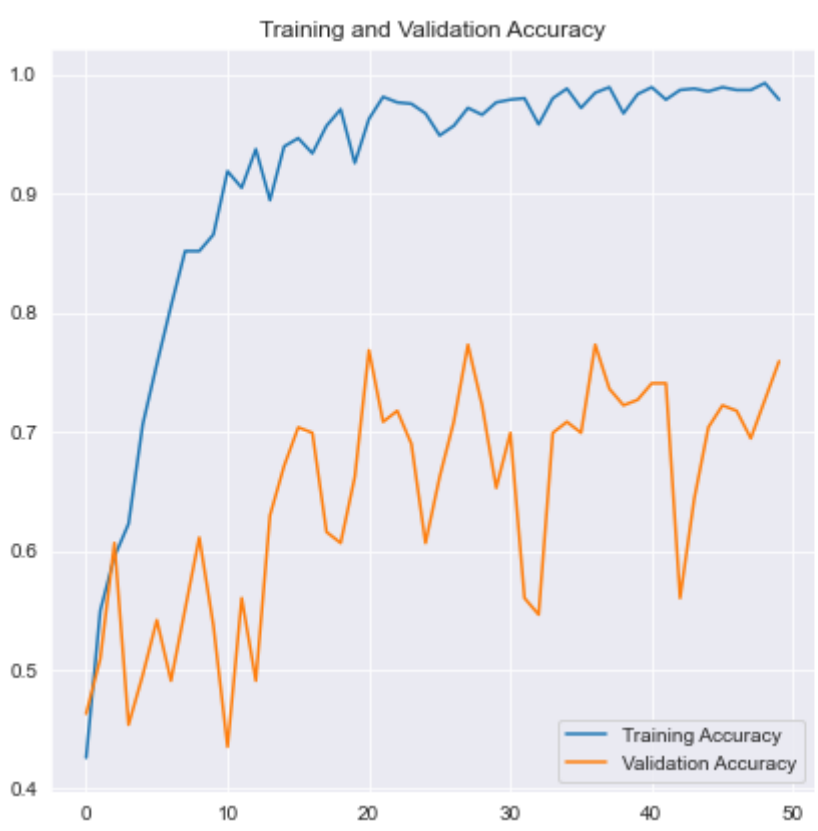
In [35]:
```
k=1
j=0
for i in range(0,250,50):
    j +=50
    print('Plot for ',k,'cross validation accuracy and loss for Training and Validation phase')
    k +=1
    plot_print(i,j)
```

Plot for 1 cross validation accuracy and loss for Training and Validation phase

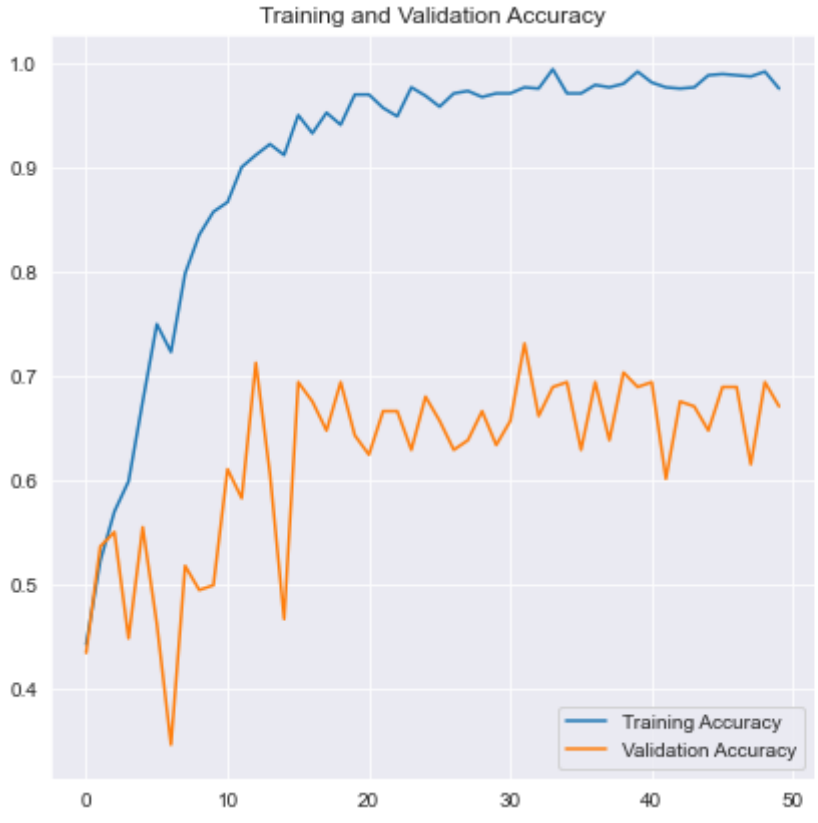




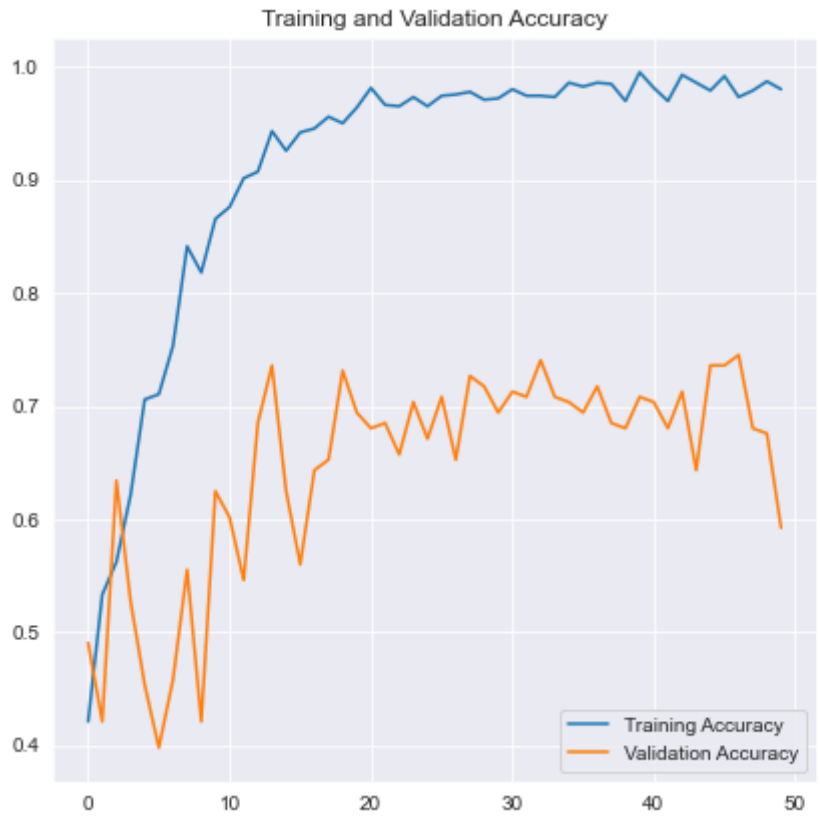
Plot for 2 cross validation accuracy and loss for Training and Validation phase



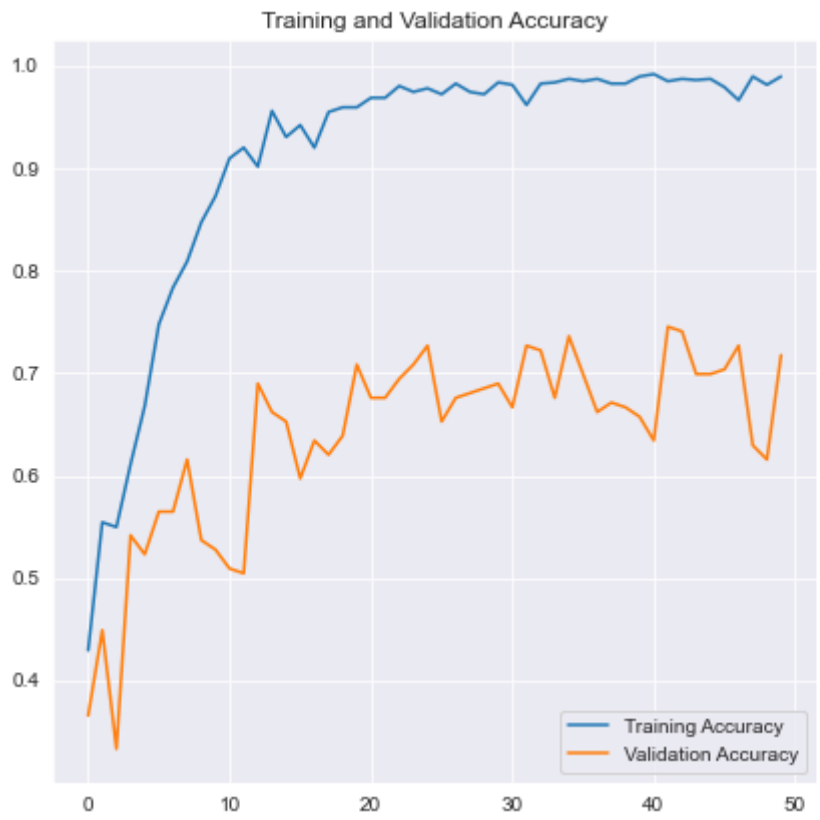
Plot for 3 cross validation accuracy and loss for Training and Validation phase



Plot for 4 cross validation accuracy and loss for Training and Validation phase



Plot for 5 cross validation accuracy and loss for Training and Validation phase



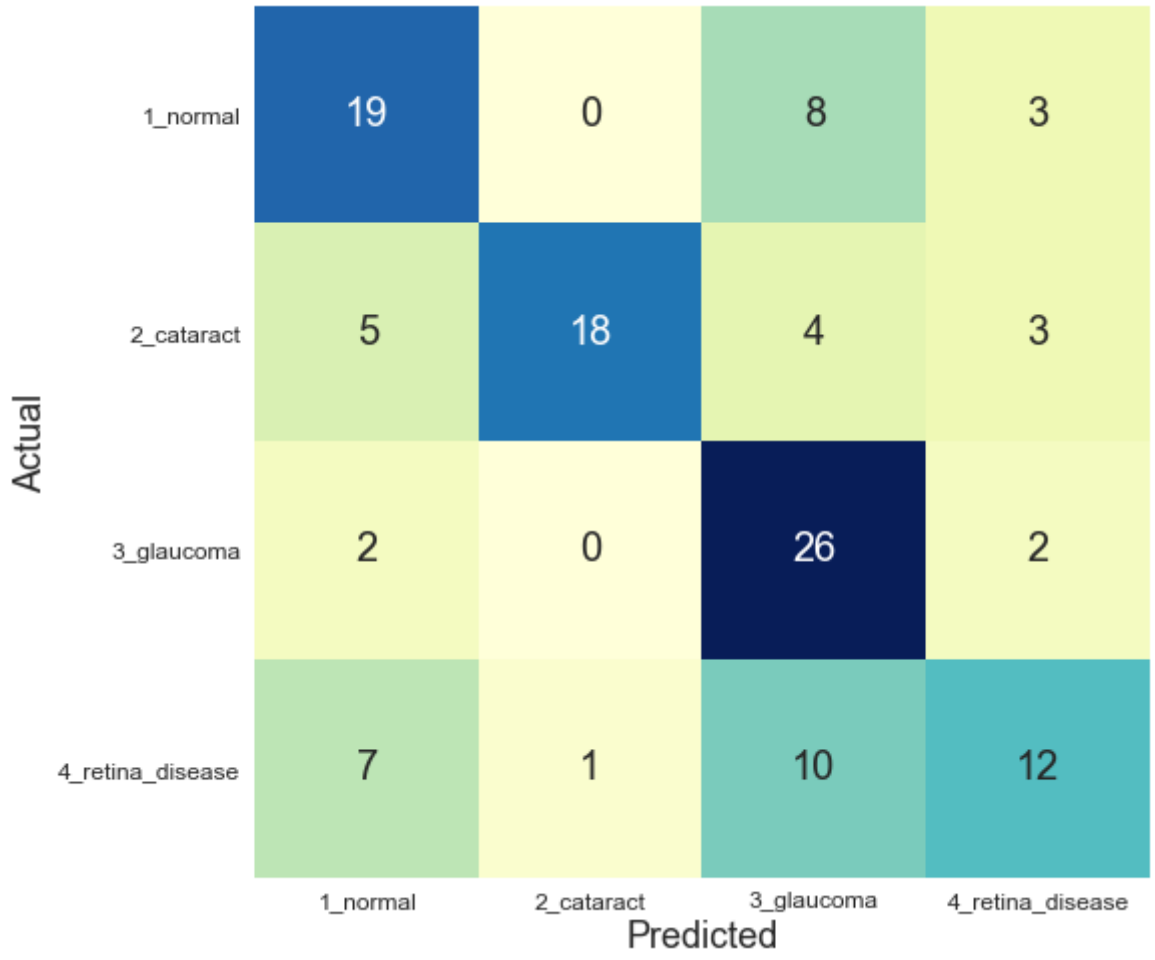
Visualizing Confusion Matrix for Each Fold

In [36]:
```
CM= np.array(CM)
CM.resize(5,4,4)
```

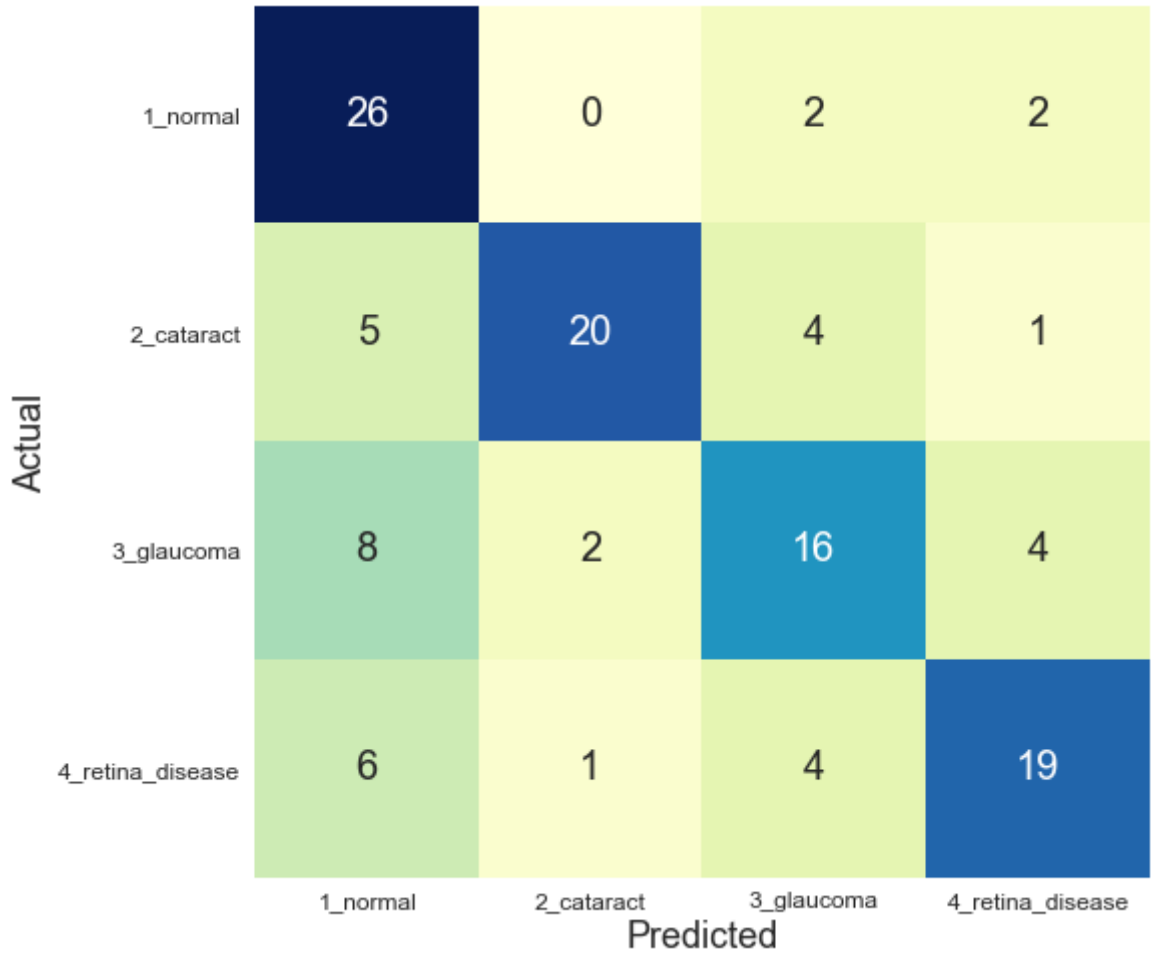
In [37]:
```
def confusionmatrix_vis(i):

        yticklabels=['1_normal', '2_cataract','3_glaucoma','4_retina_disease']
        xticklabels=['1_normal', '2_cataract','3_glaucoma','4_retina_disease']
        plt.figure(figsize=(8, 8))
        hm =sns.heatmap(CM[i], annot=True,annot_kws={"size": 20}, cbar=False,cmap="YlGnBu",yticklabels=yticklabels,xti
cklabels=xticklabels)

        hm.set_xticklabels(hm.get_xticklabels(), rotation=0, fontsize = 12, )
        hm.set_yticklabels(hm.get_yticklabels(), rotation=0, fontsize = 12)

        plt.ylabel("Actual", fontsize = 18)
        plt.xlabel("Predicted",fontsize = 18)

        return plt.show()
```

In [38]:
```
k=1
for i in range(5):
    print('Confusion Matrix for ',k,'Cross Validation Test phase')
    k +=1
    confusionmatrix_vis(i)
```

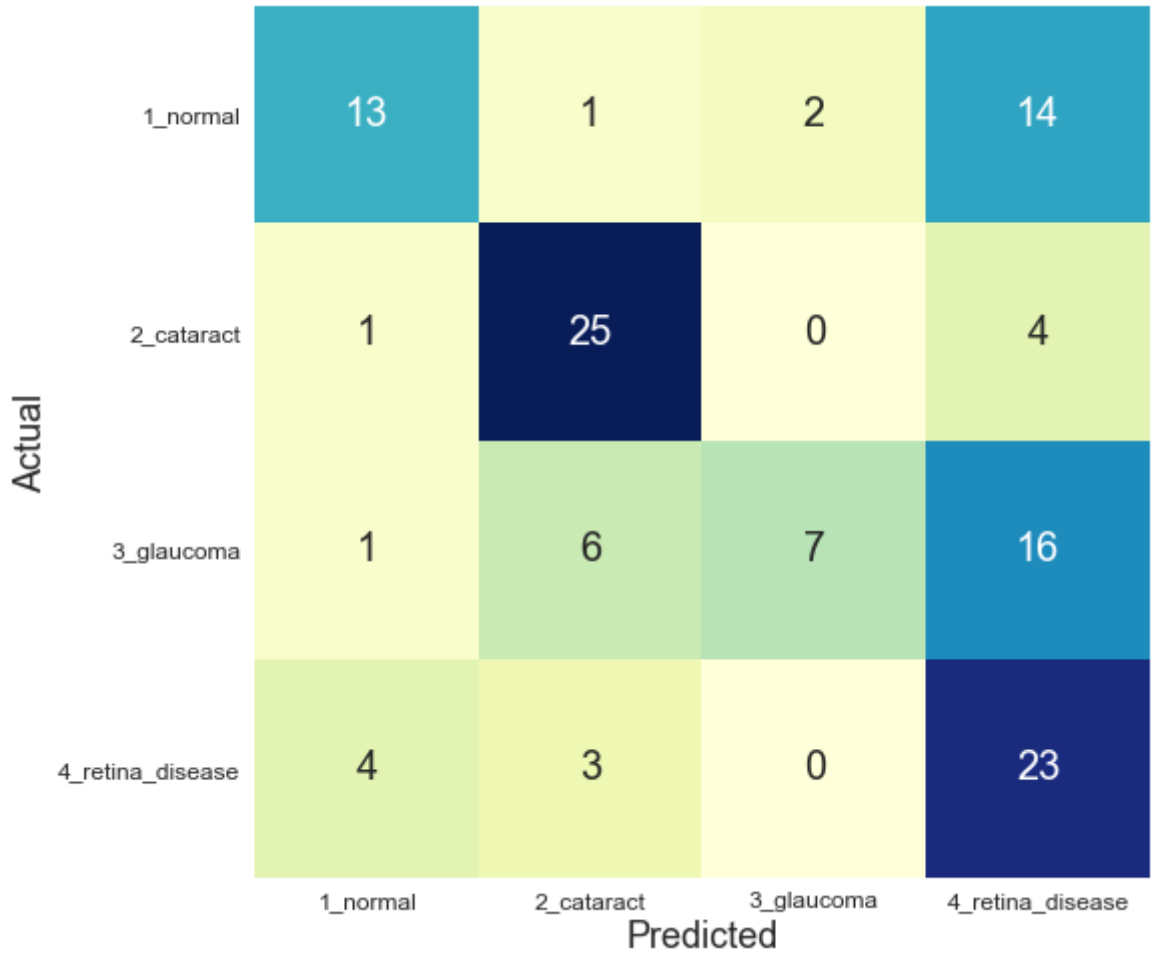
Confusion Matrix for 1 Cross Validation Test phase



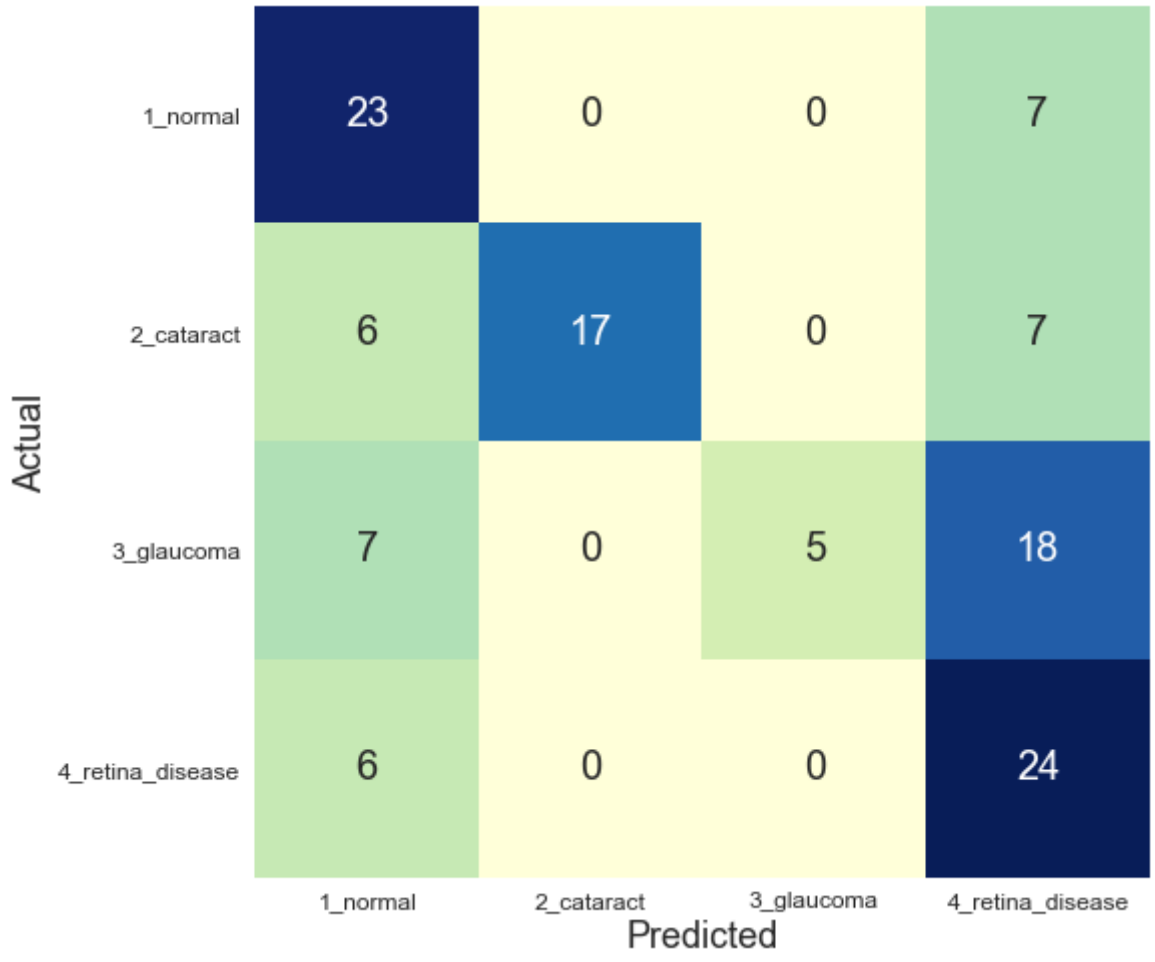
Confusion Matrix for 2 Cross Validation Test phase



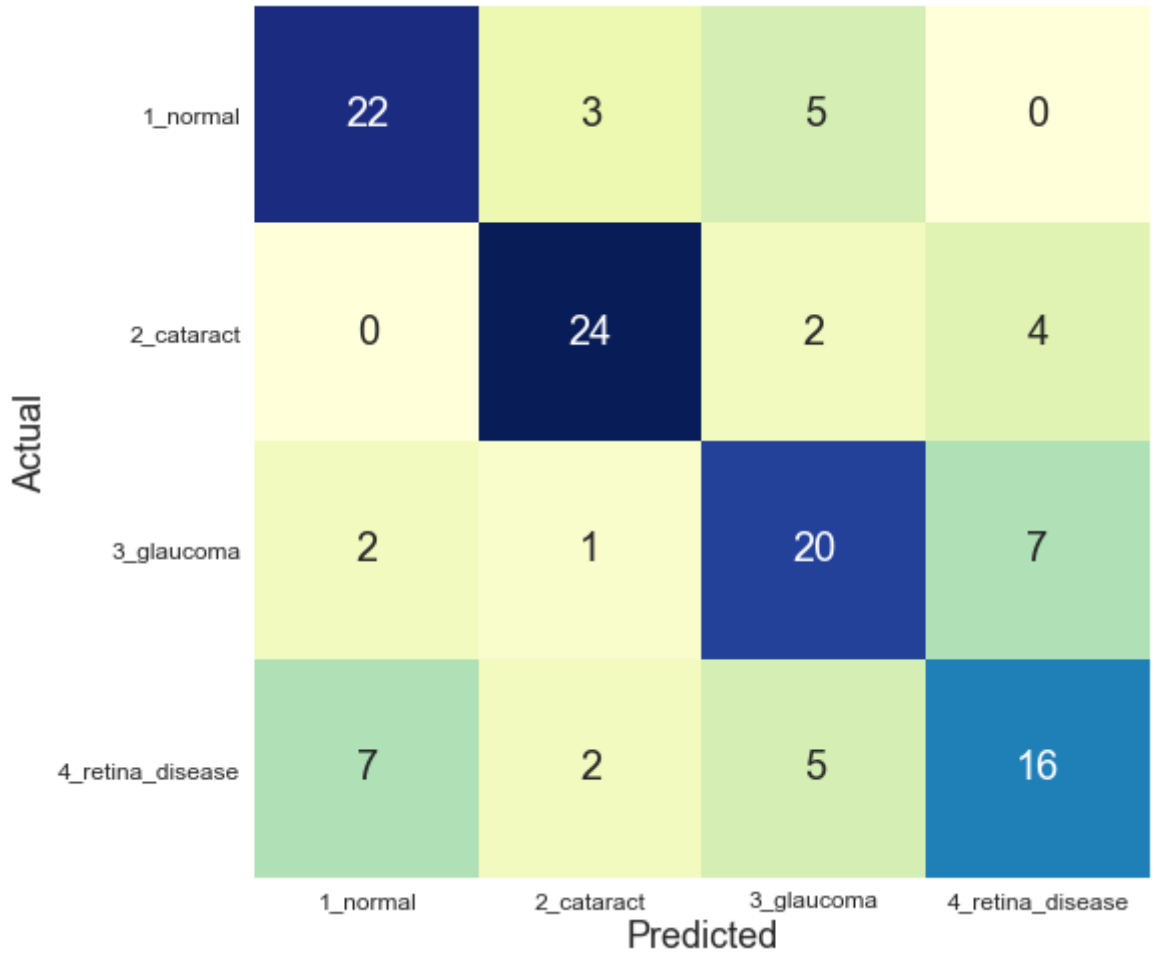
Confusion Matrix for 3 Cross Validation Test phase



Confusion Matrix for 4 Cross Validation Test phase



Confusion Matrix for 5 Cross Validation Test phase

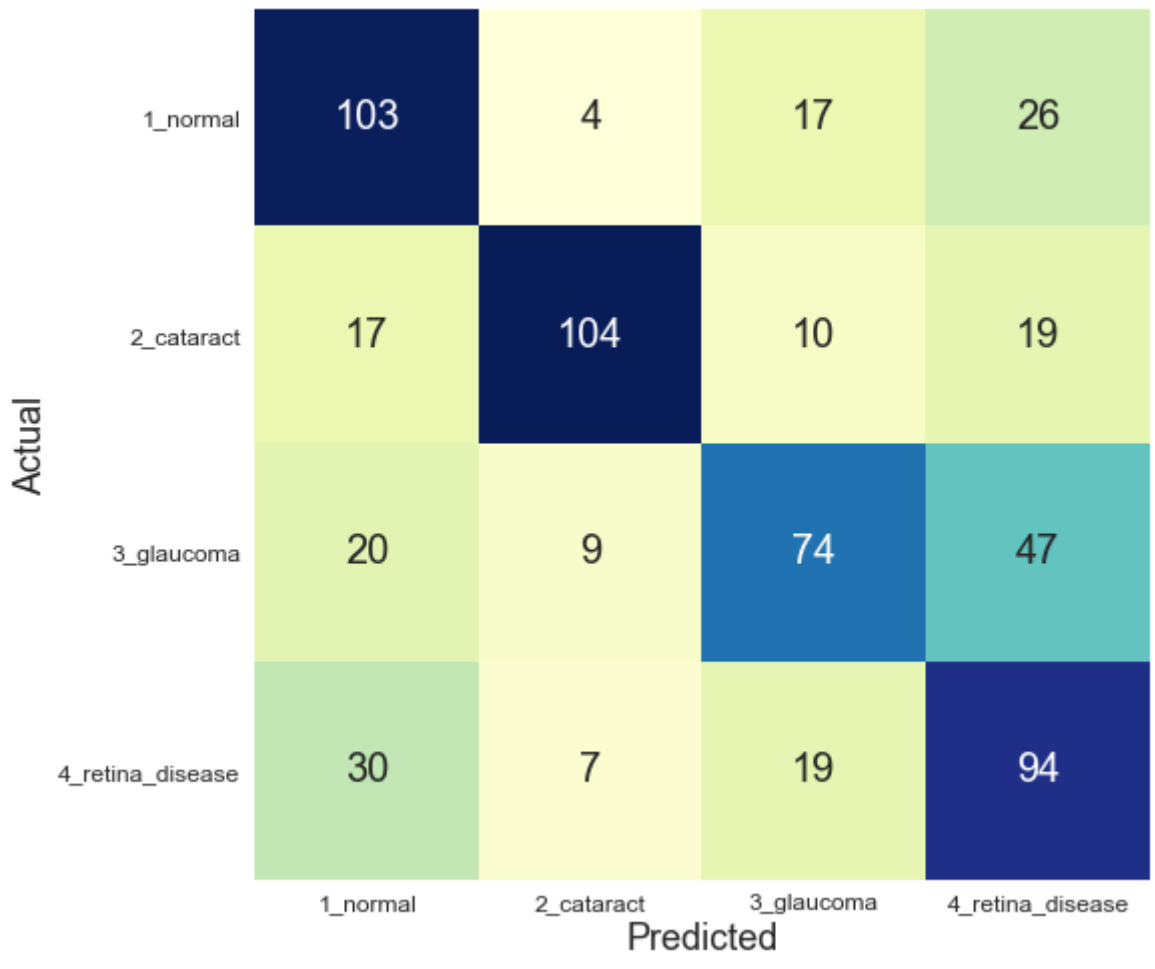


Visualizing Summarized Confusion Matrix of all 5 folds

In [39]:
```
CM_sum = CM[0]+CM[1]+CM[2]+CM[3]+CM[4]
CM_sum
```

Out[39]:
```
array([[103,   4,  17,  26],
       [ 17, 104,  10,  19],
       [ 20,   9,  74,  47],
       [ 30,   7,  19,  94]], dtype=int64)
```

In [40]:
```
yticklabels=['1_normal', '2_cataract','3_glaucoma','4_retina_disease']
xticklabels=['1_normal', '2_cataract','3_glaucoma','4_retina_disease']
plt.figure(figsize=(8, 8))
hm =sns.heatmap(CM_sum, annot=True,annot_kws={"size": 20},fmt='g', cbar=False,cmap="YlGnBu",yticklabels=yticklabels,xt
icklabels=xticklabels)

hm.set_xticklabels(hm.get_xticklabels(), rotation=0, fontsize = 12, )
hm.set_yticklabels(hm.get_yticklabels(), rotation=0, fontsize = 12)

plt.ylabel("Actual", fontsize = 18)
plt.xlabel("Predicted",fontsize = 18)

plt.show()
```



Reconfirming the values of Accuracy,Sensitivity and Specificity

In [41]:
```
    sensitivity_1_normal = (CM_sum[0,0])/(CM_sum[0,0]+CM_sum[0,1]+CM_sum[0,2]+CM_sum[0,3])
    #print('Sensitivity_1_normal          : ', sensitivity_1_normal )

    sensitivity_2_cataract = (CM_sum[1,1])/(CM_sum[1,0]+CM_sum[1,1]+CM_sum[1,2]+CM_sum[1,3])
    #print('Sensitivity_2_cataract        : ', sensitivity_2_cataract )


    sensitivity_3_glaucoma = (CM_sum[2,2])/(CM_sum[2,0]+CM_sum[2,1]+CM_sum[2,2]+CM_sum[2,3])
    #print('Sensitivity_3_glaucoma        : ', sensitivity_3_glaucoma )

    sensitivity_4_retina_disease = (CM_sum[3,3])/(CM_sum[3,0]+CM_sum[3,1]+CM_sum[3,2]+CM_sum[3,3])
    #print('Sensitivity_4_retina_disease  : ', sensitivity_4_retina_disease )

    specificity_1_normal = (CM_sum[1,1]+CM_sum[1,2]+CM_sum[1,3]+CM_sum[2,1]+CM_sum[2,2]+CM_sum[2,3]+CM_sum[3,1]+CM_sum
[3,2]+CM_sum[3,3])/(CM_sum[1,0]+CM_sum[2,0]+CM_sum[3,0]+CM_sum[1,1]+CM_sum[1,2]+CM_sum[1,3]+CM_sum[2,1]+CM_sum[2,2]+CM
_sum[2,3]+CM_sum[3,1]+CM_sum[3,2]+CM_sum[3,3])
    #print('Specificity : ', specificity_1_normal)

    specificity_2_cataract = (CM_sum[0,0]+CM_sum[0,2]+CM_sum[0,3]+CM_sum[2,0]+CM_sum[2,2]+CM_sum[2,3]+CM_sum[3,0]+CM_s
um[3,2]+CM_sum[3,3])/(CM_sum[0,1]+CM_sum[2,1]+CM_sum[3,1]+CM_sum[0,0]+CM_sum[0,2]+CM_sum[0,3]+CM_sum[2,0]+CM_sum[2,2]+
CM_sum[2,3]+CM_sum[3,0]+CM_sum[3,2]+CM_sum[3,3])
    #print('Specificity : ', specificity_2_cataract)

    specificity_3_glaucoma = (CM_sum[0,0]+CM_sum[0,1]+CM_sum[0,3]+CM_sum[1,0]+CM_sum[1,1]+CM_sum[1,3]+CM_sum[3,0]+CM_s
um[3,1]+CM_sum[3,3])/(CM_sum[0,2]+CM_sum[1,2]+CM_sum[3,2]+CM_sum[0,0]+CM_sum[0,1]+CM_sum[0,3]+CM_sum[1,0]+CM_sum[1,1]+
CM_sum[1,3]+CM_sum[3,0]+CM_sum[3,1]+CM_sum[3,3])
    #print('Specificity : ', specificity_3_glaucoma)

    specificity_4_retina_disease= (CM_sum[0,0]+CM_sum[0,1]+CM_sum[0,2]+CM_sum[1,0]+CM_sum[1,1]+CM_sum[1,2]+CM_sum[2,0]
+CM_sum[2,1]+CM_sum[2,2])/(CM_sum[0,3]+CM_sum[1,3]+CM_sum[2,3]+CM_sum[0,0]+CM_sum[0,1]+CM_sum[0,2]+CM_sum[1,0]+CM_sum[
1,1]+CM_sum[1,2]+CM_sum[2,0]+CM_sum[2,1]+CM_sum[2,2])
    #print('Specificity : ', specificity_4_retina_disease)

    Sensitivity= (sensitivity_1_normal + sensitivity_2_cataract + sensitivity_3_glaucoma + sensitivity_4_retina_diseas
e)/4
    #print(Sensitivity)

    Specificity= (specificity_1_normal + specificity_2_cataract + specificity_3_glaucoma + specificity_4_retina_diseas
e)/4
    #print(Specificity)

    total1=sum(sum(CM_sum))
    test_accuracy=(CM_sum[0,0]+CM_sum[1,1]+CM_sum[2,2]+CM_sum[3,3])/total1

    print ('Accuracy    : ', test_accuracy)
    print ('Specificity : ', Specificity)
    print ('Sensitivity : ', Sensitivity)
```

```
Accuracy    :  0.625
Specificity :  0.8386044311512217
Sensitivity :  0.625
```

Model Summary

In [42]:
```
model_build_compile(k)
```

```
model building and compiling for fold 7
```

Out[42]:
```
<tensorflow.python.keras.engine.functional.Functional at 0x2a582772df0>
```

In [43]:
```
model.summary()
```

Model: "model_4"

Layer (type)	Output Shape	Param #	Connected to
input_5 (InputLayer)	[(None, 224, 224, 3)	0	
conv2d_812 (Conv2D)	(None, 111, 111, 32)	864	input_5[0][0]
batch_normalization_824 (BatchN	(None, 111, 111, 32)	96	conv2d_812[0][0]
activation_812 (Activation)	(None, 111, 111, 32)	0	batch_normalization_824[0][0]
conv2d_813 (Conv2D)	(None, 109, 109, 32)	9216	activation_812[0][0]
batch_normalization_825 (BatchN	(None, 109, 109, 32)	96	conv2d_813[0][0]
activation_813 (Activation)	(None, 109, 109, 32)	0	batch_normalization_825[0][0]
conv2d_814 (Conv2D)	(None, 109, 109, 64)	18432	activation_813[0][0]
batch_normalization_826 (BatchN	(None, 109, 109, 64)	192	conv2d_814[0][0]
activation_814 (Activation)	(None, 109, 109, 64)	0	batch_normalization_826[0][0]
max_pooling2d_16 (MaxPooling2D)	(None, 54, 54, 64)	0	activation_814[0][0]
conv2d_815 (Conv2D)	(None, 54, 54, 80)	5120	max_pooling2d_16[0][0]
batch_normalization_827 (BatchN	(None, 54, 54, 80)	240	conv2d_815[0][0]
activation_815 (Activation)	(None, 54, 54, 80)	0	batch_normalization_827[0][0]
conv2d_816 (Conv2D)	(None, 52, 52, 192)	138240	activation_815[0][0]
batch_normalization_828 (BatchN	(None, 52, 52, 192)	576	conv2d_816[0][0]
activation_816 (Activation)	(None, 52, 52, 192)	0	batch_normalization_828[0][0]
max_pooling2d_17 (MaxPooling2D)	(None, 25, 25, 192)	0	activation_816[0][0]
conv2d_820 (Conv2D)	(None, 25, 25, 64)	12288	max_pooling2d_17[0][0]
batch_normalization_832 (BatchN	(None, 25, 25, 64)	192	conv2d_820[0][0]
activation_820 (Activation)	(None, 25, 25, 64)	0	batch_normalization_832[0][0]
conv2d_818 (Conv2D)	(None, 25, 25, 48)	9216	max_pooling2d_17[0][0]
conv2d_821 (Conv2D)	(None, 25, 25, 96)	55296	activation_820[0][0]
batch_normalization_830 (BatchN	(None, 25, 25, 48)	144	conv2d_818[0][0]
batch_normalization_833 (BatchN	(None, 25, 25, 96)	288	conv2d_821[0][0]
activation_818 (Activation)	(None, 25, 25, 48)	0	batch_normalization_830[0][0]
activation_821 (Activation)	(None, 25, 25, 96)	0	batch_normalization_833[0][0]
average_pooling2d_4 (AveragePoo	(None, 25, 25, 192)	0	max_pooling2d_17[0][0]
conv2d_817 (Conv2D)	(None, 25, 25, 96)	18432	max_pooling2d_17[0][0]
conv2d_819 (Conv2D)	(None, 25, 25, 64)	76800	activation_818[0][0]
conv2d_822 (Conv2D)	(None, 25, 25, 96)	82944	activation_821[0][0]
conv2d_823 (Conv2D)	(None, 25, 25, 64)	12288	average_pooling2d_4[0][0]
batch_normalization_829 (BatchN	(None, 25, 25, 96)	288	conv2d_817[0][0]
batch_normalization_831 (BatchN	(None, 25, 25, 64)	192	conv2d_819[0][0]
batch_normalization_834 (BatchN	(None, 25, 25, 96)	288	conv2d_822[0][0]
batch_normalization_835 (BatchN	(None, 25, 25, 64)	192	conv2d_823[0][0]
activation_817 (Activation)	(None, 25, 25, 96)	0	batch_normalization_829[0][0]
activation_819 (Activation)	(None, 25, 25, 64)	0	batch_normalization_831[0][0]
activation_822 (Activation)	(None, 25, 25, 96)	0	batch_normalization_834[0][0]
activation_823 (Activation)	(None, 25, 25, 64)	0	batch_normalization_835[0][0]
mixed_5b (Concatenate)	(None, 25, 25, 320)	0	activation_817[0][0] activation_819[0][0] activation_822[0][0] activation_823[0][0]

conv2d_827 (Conv2D)	(None, 25, 25, 32)	10240	mixed_5b[0][0]
batch_normalization_839 (BatchN	(None, 25, 25, 32)	96	conv2d_827[0][0]
activation_827 (Activation)	(None, 25, 25, 32)	0	batch_normalization_839[0][0]
conv2d_825 (Conv2D)	(None, 25, 25, 32)	10240	mixed_5b[0][0]
conv2d_828 (Conv2D)	(None, 25, 25, 48)	13824	activation_827[0][0]
batch_normalization_837 (BatchN	(None, 25, 25, 32)	96	conv2d_825[0][0]
batch_normalization_840 (BatchN	(None, 25, 25, 48)	144	conv2d_828[0][0]
activation_825 (Activation)	(None, 25, 25, 32)	0	batch_normalization_837[0][0]
activation_828 (Activation)	(None, 25, 25, 48)	0	batch_normalization_840[0][0]
conv2d_824 (Conv2D)	(None, 25, 25, 32)	10240	mixed_5b[0][0]
conv2d_826 (Conv2D)	(None, 25, 25, 32)	9216	activation_825[0][0]
conv2d_829 (Conv2D)	(None, 25, 25, 64)	27648	activation_828[0][0]
batch_normalization_836 (BatchN	(None, 25, 25, 32)	96	conv2d_824[0][0]
batch_normalization_838 (BatchN	(None, 25, 25, 32)	96	conv2d_826[0][0]
batch_normalization_841 (BatchN	(None, 25, 25, 64)	192	conv2d_829[0][0]
activation_824 (Activation)	(None, 25, 25, 32)	0	batch_normalization_836[0][0]
activation_826 (Activation)	(None, 25, 25, 32)	0	batch_normalization_838[0][0]
activation_829 (Activation)	(None, 25, 25, 64)	0	batch_normalization_841[0][0]
block35_1_mixed (Concatenate)	(None, 25, 25, 128)	0	activation_824[0][0] activation_826[0][0] activation_829[0][0]
block35_1_conv (Conv2D)	(None, 25, 25, 320)	41280	block35_1_mixed[0][0]
block35_1 (Lambda)	(None, 25, 25, 320)	0	mixed_5b[0][0] block35_1_conv[0][0]
block35_1_ac (Activation)	(None, 25, 25, 320)	0	block35_1[0][0]
conv2d_833 (Conv2D)	(None, 25, 25, 32)	10240	block35_1_ac[0][0]
batch_normalization_845 (BatchN	(None, 25, 25, 32)	96	conv2d_833[0][0]
activation_833 (Activation)	(None, 25, 25, 32)	0	batch_normalization_845[0][0]
conv2d_831 (Conv2D)	(None, 25, 25, 32)	10240	block35_1_ac[0][0]
conv2d_834 (Conv2D)	(None, 25, 25, 48)	13824	activation_833[0][0]
batch_normalization_843 (BatchN	(None, 25, 25, 32)	96	conv2d_831[0][0]
batch_normalization_846 (BatchN	(None, 25, 25, 48)	144	conv2d_834[0][0]
activation_831 (Activation)	(None, 25, 25, 32)	0	batch_normalization_843[0][0]
activation_834 (Activation)	(None, 25, 25, 48)	0	batch_normalization_846[0][0]
conv2d_830 (Conv2D)	(None, 25, 25, 32)	10240	block35_1_ac[0][0]
conv2d_832 (Conv2D)	(None, 25, 25, 32)	9216	activation_831[0][0]
conv2d_835 (Conv2D)	(None, 25, 25, 64)	27648	activation_834[0][0]
batch_normalization_842 (BatchN	(None, 25, 25, 32)	96	conv2d_830[0][0]
batch_normalization_844 (BatchN	(None, 25, 25, 32)	96	conv2d_832[0][0]
batch_normalization_847 (BatchN	(None, 25, 25, 64)	192	conv2d_835[0][0]
activation_830 (Activation)	(None, 25, 25, 32)	0	batch_normalization_842[0][0]
activation_832 (Activation)	(None, 25, 25, 32)	0	batch_normalization_844[0][0]
activation_835 (Activation)	(None, 25, 25, 64)	0	batch_normalization_847[0][0]
block35_2_mixed (Concatenate)	(None, 25, 25, 128)	0	activation_830[0][0] activation_832[0][0] activation_835[0][0]

block35_2_conv (Conv2D)	(None, 25, 25, 320)	41280	block35_2_mixed[0][0]
block35_2 (Lambda)	(None, 25, 25, 320)	0	block35_1_ac[0][0] block35_2_conv[0][0]
block35_2_ac (Activation)	(None, 25, 25, 320)	0	block35_2[0][0]
conv2d_839 (Conv2D)	(None, 25, 25, 32)	10240	block35_2_ac[0][0]
batch_normalization_851 (BatchN	(None, 25, 25, 32)	96	conv2d_839[0][0]
activation_839 (Activation)	(None, 25, 25, 32)	0	batch_normalization_851[0][0]
conv2d_837 (Conv2D)	(None, 25, 25, 32)	10240	block35_2_ac[0][0]
conv2d_840 (Conv2D)	(None, 25, 25, 48)	13824	activation_839[0][0]
batch_normalization_849 (BatchN	(None, 25, 25, 32)	96	conv2d_837[0][0]
batch_normalization_852 (BatchN	(None, 25, 25, 48)	144	conv2d_840[0][0]
activation_837 (Activation)	(None, 25, 25, 32)	0	batch_normalization_849[0][0]
activation_840 (Activation)	(None, 25, 25, 48)	0	batch_normalization_852[0][0]
conv2d_836 (Conv2D)	(None, 25, 25, 32)	10240	block35_2_ac[0][0]
conv2d_838 (Conv2D)	(None, 25, 25, 32)	9216	activation_837[0][0]
conv2d_841 (Conv2D)	(None, 25, 25, 64)	27648	activation_840[0][0]
batch_normalization_848 (BatchN	(None, 25, 25, 32)	96	conv2d_836[0][0]
batch_normalization_850 (BatchN	(None, 25, 25, 32)	96	conv2d_838[0][0]
batch_normalization_853 (BatchN	(None, 25, 25, 64)	192	conv2d_841[0][0]
activation_836 (Activation)	(None, 25, 25, 32)	0	batch_normalization_848[0][0]
activation_838 (Activation)	(None, 25, 25, 32)	0	batch_normalization_850[0][0]
activation_841 (Activation)	(None, 25, 25, 64)	0	batch_normalization_853[0][0]
block35_3_mixed (Concatenate)	(None, 25, 25, 128)	0	activation_836[0][0] activation_838[0][0] activation_841[0][0]
block35_3_conv (Conv2D)	(None, 25, 25, 320)	41280	block35_3_mixed[0][0]
block35_3 (Lambda)	(None, 25, 25, 320)	0	block35_2_ac[0][0] block35_3_conv[0][0]
block35_3_ac (Activation)	(None, 25, 25, 320)	0	block35_3[0][0]
conv2d_845 (Conv2D)	(None, 25, 25, 32)	10240	block35_3_ac[0][0]
batch_normalization_857 (BatchN	(None, 25, 25, 32)	96	conv2d_845[0][0]
activation_845 (Activation)	(None, 25, 25, 32)	0	batch_normalization_857[0][0]
conv2d_843 (Conv2D)	(None, 25, 25, 32)	10240	block35_3_ac[0][0]
conv2d_846 (Conv2D)	(None, 25, 25, 48)	13824	activation_845[0][0]
batch_normalization_855 (BatchN	(None, 25, 25, 32)	96	conv2d_843[0][0]
batch_normalization_858 (BatchN	(None, 25, 25, 48)	144	conv2d_846[0][0]
activation_843 (Activation)	(None, 25, 25, 32)	0	batch_normalization_855[0][0]
activation_846 (Activation)	(None, 25, 25, 48)	0	batch_normalization_858[0][0]
conv2d_842 (Conv2D)	(None, 25, 25, 32)	10240	block35_3_ac[0][0]
conv2d_844 (Conv2D)	(None, 25, 25, 32)	9216	activation_843[0][0]
conv2d_847 (Conv2D)	(None, 25, 25, 64)	27648	activation_846[0][0]
batch_normalization_854 (BatchN	(None, 25, 25, 32)	96	conv2d_842[0][0]
batch_normalization_856 (BatchN	(None, 25, 25, 32)	96	conv2d_844[0][0]
batch_normalization_859 (BatchN	(None, 25, 25, 64)	192	conv2d_847[0][0]
activation_842 (Activation)	(None, 25, 25, 32)	0	batch_normalization_854[0][0]
activation_844 (Activation)	(None, 25, 25, 32)	0	batch_normalization_856[0][0]

activation_847 (Activation)	(None, 25, 25, 64)	0	batch_normalization_859[0][0]
block35_4_mixed (Concatenate)	(None, 25, 25, 128)	0	activation_842[0][0] activation_844[0][0] activation_847[0][0]
block35_4_conv (Conv2D)	(None, 25, 25, 320)	41280	block35_4_mixed[0][0]
block35_4 (Lambda)	(None, 25, 25, 320)	0	block35_3_ac[0][0] block35_4_conv[0][0]
block35_4_ac (Activation)	(None, 25, 25, 320)	0	block35_4[0][0]
conv2d_851 (Conv2D)	(None, 25, 25, 32)	10240	block35_4_ac[0][0]
batch_normalization_863 (BatchN	(None, 25, 25, 32)	96	conv2d_851[0][0]
activation_851 (Activation)	(None, 25, 25, 32)	0	batch_normalization_863[0][0]
conv2d_849 (Conv2D)	(None, 25, 25, 32)	10240	block35_4_ac[0][0]
conv2d_852 (Conv2D)	(None, 25, 25, 48)	13824	activation_851[0][0]
batch_normalization_861 (BatchN	(None, 25, 25, 32)	96	conv2d_849[0][0]
batch_normalization_864 (BatchN	(None, 25, 25, 48)	144	conv2d_852[0][0]
activation_849 (Activation)	(None, 25, 25, 32)	0	batch_normalization_861[0][0]
activation_852 (Activation)	(None, 25, 25, 48)	0	batch_normalization_864[0][0]
conv2d_848 (Conv2D)	(None, 25, 25, 32)	10240	block35_4_ac[0][0]
conv2d_850 (Conv2D)	(None, 25, 25, 32)	9216	activation_849[0][0]
conv2d_853 (Conv2D)	(None, 25, 25, 64)	27648	activation_852[0][0]
batch_normalization_860 (BatchN	(None, 25, 25, 32)	96	conv2d_848[0][0]
batch_normalization_862 (BatchN	(None, 25, 25, 32)	96	conv2d_850[0][0]
batch_normalization_865 (BatchN	(None, 25, 25, 64)	192	conv2d_853[0][0]
activation_848 (Activation)	(None, 25, 25, 32)	0	batch_normalization_860[0][0]
activation_850 (Activation)	(None, 25, 25, 32)	0	batch_normalization_862[0][0]
activation_853 (Activation)	(None, 25, 25, 64)	0	batch_normalization_865[0][0]
block35_5_mixed (Concatenate)	(None, 25, 25, 128)	0	activation_848[0][0] activation_850[0][0] activation_853[0][0]
block35_5_conv (Conv2D)	(None, 25, 25, 320)	41280	block35_5_mixed[0][0]
block35_5 (Lambda)	(None, 25, 25, 320)	0	block35_4_ac[0][0] block35_5_conv[0][0]
block35_5_ac (Activation)	(None, 25, 25, 320)	0	block35_5[0][0]
conv2d_857 (Conv2D)	(None, 25, 25, 32)	10240	block35_5_ac[0][0]
batch_normalization_869 (BatchN	(None, 25, 25, 32)	96	conv2d_857[0][0]
activation_857 (Activation)	(None, 25, 25, 32)	0	batch_normalization_869[0][0]
conv2d_855 (Conv2D)	(None, 25, 25, 32)	10240	block35_5_ac[0][0]
conv2d_858 (Conv2D)	(None, 25, 25, 48)	13824	activation_857[0][0]
batch_normalization_867 (BatchN	(None, 25, 25, 32)	96	conv2d_855[0][0]
batch_normalization_870 (BatchN	(None, 25, 25, 48)	144	conv2d_858[0][0]
activation_855 (Activation)	(None, 25, 25, 32)	0	batch_normalization_867[0][0]
activation_858 (Activation)	(None, 25, 25, 48)	0	batch_normalization_870[0][0]
conv2d_854 (Conv2D)	(None, 25, 25, 32)	10240	block35_5_ac[0][0]
conv2d_856 (Conv2D)	(None, 25, 25, 32)	9216	activation_855[0][0]
conv2d_859 (Conv2D)	(None, 25, 25, 64)	27648	activation_858[0][0]
batch_normalization_866 (BatchN	(None, 25, 25, 32)	96	conv2d_854[0][0]
batch_normalization_868 (BatchN	(None, 25, 25, 32)	96	conv2d_856[0][0]

batch_normalization_871 (BatchN	(None, 25, 25, 64)	192	conv2d_859[0][0]
activation_854 (Activation)	(None, 25, 25, 32)	0	batch_normalization_866[0][0]
activation_856 (Activation)	(None, 25, 25, 32)	0	batch_normalization_868[0][0]
activation_859 (Activation)	(None, 25, 25, 64)	0	batch_normalization_871[0][0]
block35_6_mixed (Concatenate)	(None, 25, 25, 128)	0	activation_854[0][0] activation_856[0][0] activation_859[0][0]
block35_6_conv (Conv2D)	(None, 25, 25, 320)	41280	block35_6_mixed[0][0]
block35_6 (Lambda)	(None, 25, 25, 320)	0	block35_5_ac[0][0] block35_6_conv[0][0]
block35_6_ac (Activation)	(None, 25, 25, 320)	0	block35_6[0][0]
conv2d_863 (Conv2D)	(None, 25, 25, 32)	10240	block35_6_ac[0][0]
batch_normalization_875 (BatchN	(None, 25, 25, 32)	96	conv2d_863[0][0]
activation_863 (Activation)	(None, 25, 25, 32)	0	batch_normalization_875[0][0]
conv2d_861 (Conv2D)	(None, 25, 25, 32)	10240	block35_6_ac[0][0]
conv2d_864 (Conv2D)	(None, 25, 25, 48)	13824	activation_863[0][0]
batch_normalization_873 (BatchN	(None, 25, 25, 32)	96	conv2d_861[0][0]
batch_normalization_876 (BatchN	(None, 25, 25, 48)	144	conv2d_864[0][0]
activation_861 (Activation)	(None, 25, 25, 32)	0	batch_normalization_873[0][0]
activation_864 (Activation)	(None, 25, 25, 48)	0	batch_normalization_876[0][0]
conv2d_860 (Conv2D)	(None, 25, 25, 32)	10240	block35_6_ac[0][0]
conv2d_862 (Conv2D)	(None, 25, 25, 32)	9216	activation_861[0][0]
conv2d_865 (Conv2D)	(None, 25, 25, 64)	27648	activation_864[0][0]
batch_normalization_872 (BatchN	(None, 25, 25, 32)	96	conv2d_860[0][0]
batch_normalization_874 (BatchN	(None, 25, 25, 32)	96	conv2d_862[0][0]
batch_normalization_877 (BatchN	(None, 25, 25, 64)	192	conv2d_865[0][0]
activation_860 (Activation)	(None, 25, 25, 32)	0	batch_normalization_872[0][0]
activation_862 (Activation)	(None, 25, 25, 32)	0	batch_normalization_874[0][0]
activation_865 (Activation)	(None, 25, 25, 64)	0	batch_normalization_877[0][0]
block35_7_mixed (Concatenate)	(None, 25, 25, 128)	0	activation_860[0][0] activation_862[0][0] activation_865[0][0]
block35_7_conv (Conv2D)	(None, 25, 25, 320)	41280	block35_7_mixed[0][0]
block35_7 (Lambda)	(None, 25, 25, 320)	0	block35_6_ac[0][0] block35_7_conv[0][0]
block35_7_ac (Activation)	(None, 25, 25, 320)	0	block35_7[0][0]
conv2d_869 (Conv2D)	(None, 25, 25, 32)	10240	block35_7_ac[0][0]
batch_normalization_881 (BatchN	(None, 25, 25, 32)	96	conv2d_869[0][0]
activation_869 (Activation)	(None, 25, 25, 32)	0	batch_normalization_881[0][0]
conv2d_867 (Conv2D)	(None, 25, 25, 32)	10240	block35_7_ac[0][0]
conv2d_870 (Conv2D)	(None, 25, 25, 48)	13824	activation_869[0][0]
batch_normalization_879 (BatchN	(None, 25, 25, 32)	96	conv2d_867[0][0]
batch_normalization_882 (BatchN	(None, 25, 25, 48)	144	conv2d_870[0][0]
activation_867 (Activation)	(None, 25, 25, 32)	0	batch_normalization_879[0][0]
activation_870 (Activation)	(None, 25, 25, 48)	0	batch_normalization_882[0][0]
conv2d_866 (Conv2D)	(None, 25, 25, 32)	10240	block35_7_ac[0][0]
conv2d_868 (Conv2D)	(None, 25, 25, 32)	9216	activation_867[0][0]

conv2d_871 (Conv2D)	(None, 25, 25, 64)	27648	activation_870[0][0]
batch_normalization_878 (BatchN	(None, 25, 25, 32)	96	conv2d_866[0][0]
batch_normalization_880 (BatchN	(None, 25, 25, 32)	96	conv2d_868[0][0]
batch_normalization_883 (BatchN	(None, 25, 25, 64)	192	conv2d_871[0][0]
activation_866 (Activation)	(None, 25, 25, 32)	0	batch_normalization_878[0][0]
activation_868 (Activation)	(None, 25, 25, 32)	0	batch_normalization_880[0][0]
activation_871 (Activation)	(None, 25, 25, 64)	0	batch_normalization_883[0][0]
block35_8_mixed (Concatenate)	(None, 25, 25, 128)	0	activation_866[0][0] activation_868[0][0] activation_871[0][0]
block35_8_conv (Conv2D)	(None, 25, 25, 320)	41280	block35_8_mixed[0][0]
block35_8 (Lambda)	(None, 25, 25, 320)	0	block35_7_ac[0][0] block35_8_conv[0][0]
block35_8_ac (Activation)	(None, 25, 25, 320)	0	block35_8[0][0]
conv2d_875 (Conv2D)	(None, 25, 25, 32)	10240	block35_8_ac[0][0]
batch_normalization_887 (BatchN	(None, 25, 25, 32)	96	conv2d_875[0][0]
activation_875 (Activation)	(None, 25, 25, 32)	0	batch_normalization_887[0][0]
conv2d_873 (Conv2D)	(None, 25, 25, 32)	10240	block35_8_ac[0][0]
conv2d_876 (Conv2D)	(None, 25, 25, 48)	13824	activation_875[0][0]
batch_normalization_885 (BatchN	(None, 25, 25, 32)	96	conv2d_873[0][0]
batch_normalization_888 (BatchN	(None, 25, 25, 48)	144	conv2d_876[0][0]
activation_873 (Activation)	(None, 25, 25, 32)	0	batch_normalization_885[0][0]
activation_876 (Activation)	(None, 25, 25, 48)	0	batch_normalization_888[0][0]
conv2d_872 (Conv2D)	(None, 25, 25, 32)	10240	block35_8_ac[0][0]
conv2d_874 (Conv2D)	(None, 25, 25, 32)	9216	activation_873[0][0]
conv2d_877 (Conv2D)	(None, 25, 25, 64)	27648	activation_876[0][0]
batch_normalization_884 (BatchN	(None, 25, 25, 32)	96	conv2d_872[0][0]
batch_normalization_886 (BatchN	(None, 25, 25, 32)	96	conv2d_874[0][0]
batch_normalization_889 (BatchN	(None, 25, 25, 64)	192	conv2d_877[0][0]
activation_872 (Activation)	(None, 25, 25, 32)	0	batch_normalization_884[0][0]
activation_874 (Activation)	(None, 25, 25, 32)	0	batch_normalization_886[0][0]
activation_877 (Activation)	(None, 25, 25, 64)	0	batch_normalization_889[0][0]
block35_9_mixed (Concatenate)	(None, 25, 25, 128)	0	activation_872[0][0] activation_874[0][0] activation_877[0][0]
block35_9_conv (Conv2D)	(None, 25, 25, 320)	41280	block35_9_mixed[0][0]
block35_9 (Lambda)	(None, 25, 25, 320)	0	block35_8_ac[0][0] block35_9_conv[0][0]
block35_9_ac (Activation)	(None, 25, 25, 320)	0	block35_9[0][0]
conv2d_881 (Conv2D)	(None, 25, 25, 32)	10240	block35_9_ac[0][0]
batch_normalization_893 (BatchN	(None, 25, 25, 32)	96	conv2d_881[0][0]
activation_881 (Activation)	(None, 25, 25, 32)	0	batch_normalization_893[0][0]
conv2d_879 (Conv2D)	(None, 25, 25, 32)	10240	block35_9_ac[0][0]
conv2d_882 (Conv2D)	(None, 25, 25, 48)	13824	activation_881[0][0]
batch_normalization_891 (BatchN	(None, 25, 25, 32)	96	conv2d_879[0][0]
batch_normalization_894 (BatchN	(None, 25, 25, 48)	144	conv2d_882[0][0]
activation_879 (Activation)	(None, 25, 25, 32)	0	batch_normalization_891[0][0]

activation_882 (Activation)	(None, 25, 25, 48)	0	batch_normalization_894[0][0]
conv2d_878 (Conv2D)	(None, 25, 25, 32)	10240	block35_9_ac[0][0]
conv2d_880 (Conv2D)	(None, 25, 25, 32)	9216	activation_879[0][0]
conv2d_883 (Conv2D)	(None, 25, 25, 64)	27648	activation_882[0][0]
batch_normalization_890 (BatchN	(None, 25, 25, 32)	96	conv2d_878[0][0]
batch_normalization_892 (BatchN	(None, 25, 25, 32)	96	conv2d_880[0][0]
batch_normalization_895 (BatchN	(None, 25, 25, 64)	192	conv2d_883[0][0]
activation_878 (Activation)	(None, 25, 25, 32)	0	batch_normalization_890[0][0]
activation_880 (Activation)	(None, 25, 25, 32)	0	batch_normalization_892[0][0]
activation_883 (Activation)	(None, 25, 25, 64)	0	batch_normalization_895[0][0]
block35_10_mixed (Concatenate)	(None, 25, 25, 128)	0	activation_878[0][0] activation_880[0][0] activation_883[0][0]
block35_10_conv (Conv2D)	(None, 25, 25, 320)	41280	block35_10_mixed[0][0]
block35_10 (Lambda)	(None, 25, 25, 320)	0	block35_9_ac[0][0] block35_10_conv[0][0]
block35_10_ac (Activation)	(None, 25, 25, 320)	0	block35_10[0][0]
conv2d_885 (Conv2D)	(None, 25, 25, 256)	81920	block35_10_ac[0][0]
batch_normalization_897 (BatchN	(None, 25, 25, 256)	768	conv2d_885[0][0]
activation_885 (Activation)	(None, 25, 25, 256)	0	batch_normalization_897[0][0]
conv2d_886 (Conv2D)	(None, 25, 25, 256)	589824	activation_885[0][0]
batch_normalization_898 (BatchN	(None, 25, 25, 256)	768	conv2d_886[0][0]
activation_886 (Activation)	(None, 25, 25, 256)	0	batch_normalization_898[0][0]
conv2d_884 (Conv2D)	(None, 12, 12, 384)	1105920	block35_10_ac[0][0]
conv2d_887 (Conv2D)	(None, 12, 12, 384)	884736	activation_886[0][0]
batch_normalization_896 (BatchN	(None, 12, 12, 384)	1152	conv2d_884[0][0]
batch_normalization_899 (BatchN	(None, 12, 12, 384)	1152	conv2d_887[0][0]
activation_884 (Activation)	(None, 12, 12, 384)	0	batch_normalization_896[0][0]
activation_887 (Activation)	(None, 12, 12, 384)	0	batch_normalization_899[0][0]
max_pooling2d_18 (MaxPooling2D)	(None, 12, 12, 320)	0	block35_10_ac[0][0]
mixed_6a (Concatenate)	(None, 12, 12, 1088)	0	activation_884[0][0] activation_887[0][0] max_pooling2d_18[0][0]
conv2d_889 (Conv2D)	(None, 12, 12, 128)	139264	mixed_6a[0][0]
batch_normalization_901 (BatchN	(None, 12, 12, 128)	384	conv2d_889[0][0]
activation_889 (Activation)	(None, 12, 12, 128)	0	batch_normalization_901[0][0]
conv2d_890 (Conv2D)	(None, 12, 12, 160)	143360	activation_889[0][0]
batch_normalization_902 (BatchN	(None, 12, 12, 160)	480	conv2d_890[0][0]
activation_890 (Activation)	(None, 12, 12, 160)	0	batch_normalization_902[0][0]
conv2d_888 (Conv2D)	(None, 12, 12, 192)	208896	mixed_6a[0][0]
conv2d_891 (Conv2D)	(None, 12, 12, 192)	215040	activation_890[0][0]
batch_normalization_900 (BatchN	(None, 12, 12, 192)	576	conv2d_888[0][0]
batch_normalization_903 (BatchN	(None, 12, 12, 192)	576	conv2d_891[0][0]
activation_888 (Activation)	(None, 12, 12, 192)	0	batch_normalization_900[0][0]
activation_891 (Activation)	(None, 12, 12, 192)	0	batch_normalization_903[0][0]
block17_1_mixed (Concatenate)	(None, 12, 12, 384)	0	activation_888[0][0] activation_891[0][0]

block17_1_conv (Conv2D)	(None, 12, 12, 1088)	418880	block17_1_mixed[0][0]
block17_1 (Lambda)	(None, 12, 12, 1088)	0	mixed_6a[0][0] block17_1_conv[0][0]
block17_1_ac (Activation)	(None, 12, 12, 1088)	0	block17_1[0][0]
conv2d_893 (Conv2D)	(None, 12, 12, 128)	139264	block17_1_ac[0][0]
batch_normalization_905 (BatchN	(None, 12, 12, 128)	384	conv2d_893[0][0]
activation_893 (Activation)	(None, 12, 12, 128)	0	batch_normalization_905[0][0]
conv2d_894 (Conv2D)	(None, 12, 12, 160)	143360	activation_893[0][0]
batch_normalization_906 (BatchN	(None, 12, 12, 160)	480	conv2d_894[0][0]
activation_894 (Activation)	(None, 12, 12, 160)	0	batch_normalization_906[0][0]
conv2d_892 (Conv2D)	(None, 12, 12, 192)	208896	block17_1_ac[0][0]
conv2d_895 (Conv2D)	(None, 12, 12, 192)	215040	activation_894[0][0]
batch_normalization_904 (BatchN	(None, 12, 12, 192)	576	conv2d_892[0][0]
batch_normalization_907 (BatchN	(None, 12, 12, 192)	576	conv2d_895[0][0]
activation_892 (Activation)	(None, 12, 12, 192)	0	batch_normalization_904[0][0]
activation_895 (Activation)	(None, 12, 12, 192)	0	batch_normalization_907[0][0]
block17_2_mixed (Concatenate)	(None, 12, 12, 384)	0	activation_892[0][0] activation_895[0][0]
block17_2_conv (Conv2D)	(None, 12, 12, 1088)	418880	block17_2_mixed[0][0]
block17_2 (Lambda)	(None, 12, 12, 1088)	0	block17_1_ac[0][0] block17_2_conv[0][0]
block17_2_ac (Activation)	(None, 12, 12, 1088)	0	block17_2[0][0]
conv2d_897 (Conv2D)	(None, 12, 12, 128)	139264	block17_2_ac[0][0]
batch_normalization_909 (BatchN	(None, 12, 12, 128)	384	conv2d_897[0][0]
activation_897 (Activation)	(None, 12, 12, 128)	0	batch_normalization_909[0][0]
conv2d_898 (Conv2D)	(None, 12, 12, 160)	143360	activation_897[0][0]
batch_normalization_910 (BatchN	(None, 12, 12, 160)	480	conv2d_898[0][0]
activation_898 (Activation)	(None, 12, 12, 160)	0	batch_normalization_910[0][0]
conv2d_896 (Conv2D)	(None, 12, 12, 192)	208896	block17_2_ac[0][0]
conv2d_899 (Conv2D)	(None, 12, 12, 192)	215040	activation_898[0][0]
batch_normalization_908 (BatchN	(None, 12, 12, 192)	576	conv2d_896[0][0]
batch_normalization_911 (BatchN	(None, 12, 12, 192)	576	conv2d_899[0][0]
activation_896 (Activation)	(None, 12, 12, 192)	0	batch_normalization_908[0][0]
activation_899 (Activation)	(None, 12, 12, 192)	0	batch_normalization_911[0][0]
block17_3_mixed (Concatenate)	(None, 12, 12, 384)	0	activation_896[0][0] activation_899[0][0]
block17_3_conv (Conv2D)	(None, 12, 12, 1088)	418880	block17_3_mixed[0][0]
block17_3 (Lambda)	(None, 12, 12, 1088)	0	block17_2_ac[0][0] block17_3_conv[0][0]
block17_3_ac (Activation)	(None, 12, 12, 1088)	0	block17_3[0][0]
conv2d_901 (Conv2D)	(None, 12, 12, 128)	139264	block17_3_ac[0][0]
batch_normalization_913 (BatchN	(None, 12, 12, 128)	384	conv2d_901[0][0]
activation_901 (Activation)	(None, 12, 12, 128)	0	batch_normalization_913[0][0]
conv2d_902 (Conv2D)	(None, 12, 12, 160)	143360	activation_901[0][0]
batch_normalization_914 (BatchN	(None, 12, 12, 160)	480	conv2d_902[0][0]
activation_902 (Activation)	(None, 12, 12, 160)	0	batch_normalization_914[0][0]
conv2d_900 (Conv2D)	(None, 12, 12, 192)	208896	block17_3_ac[0][0]

conv2d_903 (Conv2D)	(None, 12, 12, 192)	215040	activation_902[0][0]
batch_normalization_912 (BatchN	(None, 12, 12, 192)	576	conv2d_900[0][0]
batch_normalization_915 (BatchN	(None, 12, 12, 192)	576	conv2d_903[0][0]
activation_900 (Activation)	(None, 12, 12, 192)	0	batch_normalization_912[0][0]
activation_903 (Activation)	(None, 12, 12, 192)	0	batch_normalization_915[0][0]
block17_4_mixed (Concatenate)	(None, 12, 12, 384)	0	activation_900[0][0] activation_903[0][0]
block17_4_conv (Conv2D)	(None, 12, 12, 1088)	418880	block17_4_mixed[0][0]
block17_4 (Lambda)	(None, 12, 12, 1088)	0	block17_3_ac[0][0] block17_4_conv[0][0]
block17_4_ac (Activation)	(None, 12, 12, 1088)	0	block17_4[0][0]
conv2d_905 (Conv2D)	(None, 12, 12, 128)	139264	block17_4_ac[0][0]
batch_normalization_917 (BatchN	(None, 12, 12, 128)	384	conv2d_905[0][0]
activation_905 (Activation)	(None, 12, 12, 128)	0	batch_normalization_917[0][0]
conv2d_906 (Conv2D)	(None, 12, 12, 160)	143360	activation_905[0][0]
batch_normalization_918 (BatchN	(None, 12, 12, 160)	480	conv2d_906[0][0]
activation_906 (Activation)	(None, 12, 12, 160)	0	batch_normalization_918[0][0]
conv2d_904 (Conv2D)	(None, 12, 12, 192)	208896	block17_4_ac[0][0]
conv2d_907 (Conv2D)	(None, 12, 12, 192)	215040	activation_906[0][0]
batch_normalization_916 (BatchN	(None, 12, 12, 192)	576	conv2d_904[0][0]
batch_normalization_919 (BatchN	(None, 12, 12, 192)	576	conv2d_907[0][0]
activation_904 (Activation)	(None, 12, 12, 192)	0	batch_normalization_916[0][0]
activation_907 (Activation)	(None, 12, 12, 192)	0	batch_normalization_919[0][0]
block17_5_mixed (Concatenate)	(None, 12, 12, 384)	0	activation_904[0][0] activation_907[0][0]
block17_5_conv (Conv2D)	(None, 12, 12, 1088)	418880	block17_5_mixed[0][0]
block17_5 (Lambda)	(None, 12, 12, 1088)	0	block17_4_ac[0][0] block17_5_conv[0][0]
block17_5_ac (Activation)	(None, 12, 12, 1088)	0	block17_5[0][0]
conv2d_909 (Conv2D)	(None, 12, 12, 128)	139264	block17_5_ac[0][0]
batch_normalization_921 (BatchN	(None, 12, 12, 128)	384	conv2d_909[0][0]
activation_909 (Activation)	(None, 12, 12, 128)	0	batch_normalization_921[0][0]
conv2d_910 (Conv2D)	(None, 12, 12, 160)	143360	activation_909[0][0]
batch_normalization_922 (BatchN	(None, 12, 12, 160)	480	conv2d_910[0][0]
activation_910 (Activation)	(None, 12, 12, 160)	0	batch_normalization_922[0][0]
conv2d_908 (Conv2D)	(None, 12, 12, 192)	208896	block17_5_ac[0][0]
conv2d_911 (Conv2D)	(None, 12, 12, 192)	215040	activation_910[0][0]
batch_normalization_920 (BatchN	(None, 12, 12, 192)	576	conv2d_908[0][0]
batch_normalization_923 (BatchN	(None, 12, 12, 192)	576	conv2d_911[0][0]
activation_908 (Activation)	(None, 12, 12, 192)	0	batch_normalization_920[0][0]
activation_911 (Activation)	(None, 12, 12, 192)	0	batch_normalization_923[0][0]
block17_6_mixed (Concatenate)	(None, 12, 12, 384)	0	activation_908[0][0] activation_911[0][0]
block17_6_conv (Conv2D)	(None, 12, 12, 1088)	418880	block17_6_mixed[0][0]
block17_6 (Lambda)	(None, 12, 12, 1088)	0	block17_5_ac[0][0] block17_6_conv[0][0]
block17_6_ac (Activation)	(None, 12, 12, 1088)	0	block17_6[0][0]

conv2d_913 (Conv2D)	(None, 12, 12, 128)	139264	block17_6_ac[0][0]
batch_normalization_925 (BatchN	(None, 12, 12, 128)	384	conv2d_913[0][0]
activation_913 (Activation)	(None, 12, 12, 128)	0	batch_normalization_925[0][0]
conv2d_914 (Conv2D)	(None, 12, 12, 160)	143360	activation_913[0][0]
batch_normalization_926 (BatchN	(None, 12, 12, 160)	480	conv2d_914[0][0]
activation_914 (Activation)	(None, 12, 12, 160)	0	batch_normalization_926[0][0]
conv2d_912 (Conv2D)	(None, 12, 12, 192)	208896	block17_6_ac[0][0]
conv2d_915 (Conv2D)	(None, 12, 12, 192)	215040	activation_914[0][0]
batch_normalization_924 (BatchN	(None, 12, 12, 192)	576	conv2d_912[0][0]
batch_normalization_927 (BatchN	(None, 12, 12, 192)	576	conv2d_915[0][0]
activation_912 (Activation)	(None, 12, 12, 192)	0	batch_normalization_924[0][0]
activation_915 (Activation)	(None, 12, 12, 192)	0	batch_normalization_927[0][0]
block17_7_mixed (Concatenate)	(None, 12, 12, 384)	0	activation_912[0][0] activation_915[0][0]
block17_7_conv (Conv2D)	(None, 12, 12, 1088)	418880	block17_7_mixed[0][0]
block17_7 (Lambda)	(None, 12, 12, 1088)	0	block17_6_ac[0][0] block17_7_conv[0][0]
block17_7_ac (Activation)	(None, 12, 12, 1088)	0	block17_7[0][0]
conv2d_917 (Conv2D)	(None, 12, 12, 128)	139264	block17_7_ac[0][0]
batch_normalization_929 (BatchN	(None, 12, 12, 128)	384	conv2d_917[0][0]
activation_917 (Activation)	(None, 12, 12, 128)	0	batch_normalization_929[0][0]
conv2d_918 (Conv2D)	(None, 12, 12, 160)	143360	activation_917[0][0]
batch_normalization_930 (BatchN	(None, 12, 12, 160)	480	conv2d_918[0][0]
activation_918 (Activation)	(None, 12, 12, 160)	0	batch_normalization_930[0][0]
conv2d_916 (Conv2D)	(None, 12, 12, 192)	208896	block17_7_ac[0][0]
conv2d_919 (Conv2D)	(None, 12, 12, 192)	215040	activation_918[0][0]
batch_normalization_928 (BatchN	(None, 12, 12, 192)	576	conv2d_916[0][0]
batch_normalization_931 (BatchN	(None, 12, 12, 192)	576	conv2d_919[0][0]
activation_916 (Activation)	(None, 12, 12, 192)	0	batch_normalization_928[0][0]
activation_919 (Activation)	(None, 12, 12, 192)	0	batch_normalization_931[0][0]
block17_8_mixed (Concatenate)	(None, 12, 12, 384)	0	activation_916[0][0] activation_919[0][0]
block17_8_conv (Conv2D)	(None, 12, 12, 1088)	418880	block17_8_mixed[0][0]
block17_8 (Lambda)	(None, 12, 12, 1088)	0	block17_7_ac[0][0] block17_8_conv[0][0]
block17_8_ac (Activation)	(None, 12, 12, 1088)	0	block17_8[0][0]
conv2d_921 (Conv2D)	(None, 12, 12, 128)	139264	block17_8_ac[0][0]
batch_normalization_933 (BatchN	(None, 12, 12, 128)	384	conv2d_921[0][0]
activation_921 (Activation)	(None, 12, 12, 128)	0	batch_normalization_933[0][0]
conv2d_922 (Conv2D)	(None, 12, 12, 160)	143360	activation_921[0][0]
batch_normalization_934 (BatchN	(None, 12, 12, 160)	480	conv2d_922[0][0]
activation_922 (Activation)	(None, 12, 12, 160)	0	batch_normalization_934[0][0]
conv2d_920 (Conv2D)	(None, 12, 12, 192)	208896	block17_8_ac[0][0]
conv2d_923 (Conv2D)	(None, 12, 12, 192)	215040	activation_922[0][0]
batch_normalization_932 (BatchN	(None, 12, 12, 192)	576	conv2d_920[0][0]
batch_normalization_935 (BatchN	(None, 12, 12, 192)	576	conv2d_923[0][0]

activation_920 (Activation)	(None, 12, 12, 192)	0	batch_normalization_932[0][0]
activation_923 (Activation)	(None, 12, 12, 192)	0	batch_normalization_935[0][0]
block17_9_mixed (Concatenate)	(None, 12, 12, 384)	0	activation_920[0][0] activation_923[0][0]
block17_9_conv (Conv2D)	(None, 12, 12, 1088)	418880	block17_9_mixed[0][0]
block17_9 (Lambda)	(None, 12, 12, 1088)	0	block17_8_ac[0][0] block17_9_conv[0][0]
block17_9_ac (Activation)	(None, 12, 12, 1088)	0	block17_9[0][0]
conv2d_925 (Conv2D)	(None, 12, 12, 128)	139264	block17_9_ac[0][0]
batch_normalization_937 (BatchN	(None, 12, 12, 128)	384	conv2d_925[0][0]
activation_925 (Activation)	(None, 12, 12, 128)	0	batch_normalization_937[0][0]
conv2d_926 (Conv2D)	(None, 12, 12, 160)	143360	activation_925[0][0]
batch_normalization_938 (BatchN	(None, 12, 12, 160)	480	conv2d_926[0][0]
activation_926 (Activation)	(None, 12, 12, 160)	0	batch_normalization_938[0][0]
conv2d_924 (Conv2D)	(None, 12, 12, 192)	208896	block17_9_ac[0][0]
conv2d_927 (Conv2D)	(None, 12, 12, 192)	215040	activation_926[0][0]
batch_normalization_936 (BatchN	(None, 12, 12, 192)	576	conv2d_924[0][0]
batch_normalization_939 (BatchN	(None, 12, 12, 192)	576	conv2d_927[0][0]
activation_924 (Activation)	(None, 12, 12, 192)	0	batch_normalization_936[0][0]
activation_927 (Activation)	(None, 12, 12, 192)	0	batch_normalization_939[0][0]
block17_10_mixed (Concatenate)	(None, 12, 12, 384)	0	activation_924[0][0] activation_927[0][0]
block17_10_conv (Conv2D)	(None, 12, 12, 1088)	418880	block17_10_mixed[0][0]
block17_10 (Lambda)	(None, 12, 12, 1088)	0	block17_9_ac[0][0] block17_10_conv[0][0]
block17_10_ac (Activation)	(None, 12, 12, 1088)	0	block17_10[0][0]
conv2d_929 (Conv2D)	(None, 12, 12, 128)	139264	block17_10_ac[0][0]
batch_normalization_941 (BatchN	(None, 12, 12, 128)	384	conv2d_929[0][0]
activation_929 (Activation)	(None, 12, 12, 128)	0	batch_normalization_941[0][0]
conv2d_930 (Conv2D)	(None, 12, 12, 160)	143360	activation_929[0][0]
batch_normalization_942 (BatchN	(None, 12, 12, 160)	480	conv2d_930[0][0]
activation_930 (Activation)	(None, 12, 12, 160)	0	batch_normalization_942[0][0]
conv2d_928 (Conv2D)	(None, 12, 12, 192)	208896	block17_10_ac[0][0]
conv2d_931 (Conv2D)	(None, 12, 12, 192)	215040	activation_930[0][0]
batch_normalization_940 (BatchN	(None, 12, 12, 192)	576	conv2d_928[0][0]
batch_normalization_943 (BatchN	(None, 12, 12, 192)	576	conv2d_931[0][0]
activation_928 (Activation)	(None, 12, 12, 192)	0	batch_normalization_940[0][0]
activation_931 (Activation)	(None, 12, 12, 192)	0	batch_normalization_943[0][0]
block17_11_mixed (Concatenate)	(None, 12, 12, 384)	0	activation_928[0][0] activation_931[0][0]
block17_11_conv (Conv2D)	(None, 12, 12, 1088)	418880	block17_11_mixed[0][0]
block17_11 (Lambda)	(None, 12, 12, 1088)	0	block17_10_ac[0][0] block17_11_conv[0][0]
block17_11_ac (Activation)	(None, 12, 12, 1088)	0	block17_11[0][0]
conv2d_933 (Conv2D)	(None, 12, 12, 128)	139264	block17_11_ac[0][0]
batch_normalization_945 (BatchN	(None, 12, 12, 128)	384	conv2d_933[0][0]
activation_933 (Activation)	(None, 12, 12, 128)	0	batch_normalization_945[0][0]

conv2d_934 (Conv2D)	(None, 12, 12, 160)	143360	activation_933[0][0]
batch_normalization_946 (BatchN	(None, 12, 12, 160)	480	conv2d_934[0][0]
activation_934 (Activation)	(None, 12, 12, 160)	0	batch_normalization_946[0][0]
conv2d_932 (Conv2D)	(None, 12, 12, 192)	208896	block17_11_ac[0][0]
conv2d_935 (Conv2D)	(None, 12, 12, 192)	215040	activation_934[0][0]
batch_normalization_944 (BatchN	(None, 12, 12, 192)	576	conv2d_932[0][0]
batch_normalization_947 (BatchN	(None, 12, 12, 192)	576	conv2d_935[0][0]
activation_932 (Activation)	(None, 12, 12, 192)	0	batch_normalization_944[0][0]
activation_935 (Activation)	(None, 12, 12, 192)	0	batch_normalization_947[0][0]
block17_12_mixed (Concatenate)	(None, 12, 12, 384)	0	activation_932[0][0] activation_935[0][0]
block17_12_conv (Conv2D)	(None, 12, 12, 1088)	418880	block17_12_mixed[0][0]
block17_12 (Lambda)	(None, 12, 12, 1088)	0	block17_11_ac[0][0] block17_12_conv[0][0]
block17_12_ac (Activation)	(None, 12, 12, 1088)	0	block17_12[0][0]
conv2d_937 (Conv2D)	(None, 12, 12, 128)	139264	block17_12_ac[0][0]
batch_normalization_949 (BatchN	(None, 12, 12, 128)	384	conv2d_937[0][0]
activation_937 (Activation)	(None, 12, 12, 128)	0	batch_normalization_949[0][0]
conv2d_938 (Conv2D)	(None, 12, 12, 160)	143360	activation_937[0][0]
batch_normalization_950 (BatchN	(None, 12, 12, 160)	480	conv2d_938[0][0]
activation_938 (Activation)	(None, 12, 12, 160)	0	batch_normalization_950[0][0]
conv2d_936 (Conv2D)	(None, 12, 12, 192)	208896	block17_12_ac[0][0]
conv2d_939 (Conv2D)	(None, 12, 12, 192)	215040	activation_938[0][0]
batch_normalization_948 (BatchN	(None, 12, 12, 192)	576	conv2d_936[0][0]
batch_normalization_951 (BatchN	(None, 12, 12, 192)	576	conv2d_939[0][0]
activation_936 (Activation)	(None, 12, 12, 192)	0	batch_normalization_948[0][0]
activation_939 (Activation)	(None, 12, 12, 192)	0	batch_normalization_951[0][0]
block17_13_mixed (Concatenate)	(None, 12, 12, 384)	0	activation_936[0][0] activation_939[0][0]
block17_13_conv (Conv2D)	(None, 12, 12, 1088)	418880	block17_13_mixed[0][0]
block17_13 (Lambda)	(None, 12, 12, 1088)	0	block17_12_ac[0][0] block17_13_conv[0][0]
block17_13_ac (Activation)	(None, 12, 12, 1088)	0	block17_13[0][0]
conv2d_941 (Conv2D)	(None, 12, 12, 128)	139264	block17_13_ac[0][0]
batch_normalization_953 (BatchN	(None, 12, 12, 128)	384	conv2d_941[0][0]
activation_941 (Activation)	(None, 12, 12, 128)	0	batch_normalization_953[0][0]
conv2d_942 (Conv2D)	(None, 12, 12, 160)	143360	activation_941[0][0]
batch_normalization_954 (BatchN	(None, 12, 12, 160)	480	conv2d_942[0][0]
activation_942 (Activation)	(None, 12, 12, 160)	0	batch_normalization_954[0][0]
conv2d_940 (Conv2D)	(None, 12, 12, 192)	208896	block17_13_ac[0][0]
conv2d_943 (Conv2D)	(None, 12, 12, 192)	215040	activation_942[0][0]
batch_normalization_952 (BatchN	(None, 12, 12, 192)	576	conv2d_940[0][0]
batch_normalization_955 (BatchN	(None, 12, 12, 192)	576	conv2d_943[0][0]
activation_940 (Activation)	(None, 12, 12, 192)	0	batch_normalization_952[0][0]
activation_943 (Activation)	(None, 12, 12, 192)	0	batch_normalization_955[0][0]
block17_14_mixed (Concatenate)	(None, 12, 12, 384)	0	activation_940[0][0]

			activation_943[0][0]
block17_14_conv (Conv2D)	(None, 12, 12, 1088)	418880	block17_14_mixed[0][0]
block17_14 (Lambda)	(None, 12, 12, 1088)	0	block17_13_ac[0][0] block17_14_conv[0][0]
block17_14_ac (Activation)	(None, 12, 12, 1088)	0	block17_14[0][0]
conv2d_945 (Conv2D)	(None, 12, 12, 128)	139264	block17_14_ac[0][0]
batch_normalization_957 (BatchN	(None, 12, 12, 128)	384	conv2d_945[0][0]
activation_945 (Activation)	(None, 12, 12, 128)	0	batch_normalization_957[0][0]
conv2d_946 (Conv2D)	(None, 12, 12, 160)	143360	activation_945[0][0]
batch_normalization_958 (BatchN	(None, 12, 12, 160)	480	conv2d_946[0][0]
activation_946 (Activation)	(None, 12, 12, 160)	0	batch_normalization_958[0][0]
conv2d_944 (Conv2D)	(None, 12, 12, 192)	208896	block17_14_ac[0][0]
conv2d_947 (Conv2D)	(None, 12, 12, 192)	215040	activation_946[0][0]
batch_normalization_956 (BatchN	(None, 12, 12, 192)	576	conv2d_944[0][0]
batch_normalization_959 (BatchN	(None, 12, 12, 192)	576	conv2d_947[0][0]
activation_944 (Activation)	(None, 12, 12, 192)	0	batch_normalization_956[0][0]
activation_947 (Activation)	(None, 12, 12, 192)	0	batch_normalization_959[0][0]
block17_15_mixed (Concatenate)	(None, 12, 12, 384)	0	activation_944[0][0] activation_947[0][0]
block17_15_conv (Conv2D)	(None, 12, 12, 1088)	418880	block17_15_mixed[0][0]
block17_15 (Lambda)	(None, 12, 12, 1088)	0	block17_14_ac[0][0] block17_15_conv[0][0]
block17_15_ac (Activation)	(None, 12, 12, 1088)	0	block17_15[0][0]
conv2d_949 (Conv2D)	(None, 12, 12, 128)	139264	block17_15_ac[0][0]
batch_normalization_961 (BatchN	(None, 12, 12, 128)	384	conv2d_949[0][0]
activation_949 (Activation)	(None, 12, 12, 128)	0	batch_normalization_961[0][0]
conv2d_950 (Conv2D)	(None, 12, 12, 160)	143360	activation_949[0][0]
batch_normalization_962 (BatchN	(None, 12, 12, 160)	480	conv2d_950[0][0]
activation_950 (Activation)	(None, 12, 12, 160)	0	batch_normalization_962[0][0]
conv2d_948 (Conv2D)	(None, 12, 12, 192)	208896	block17_15_ac[0][0]
conv2d_951 (Conv2D)	(None, 12, 12, 192)	215040	activation_950[0][0]
batch_normalization_960 (BatchN	(None, 12, 12, 192)	576	conv2d_948[0][0]
batch_normalization_963 (BatchN	(None, 12, 12, 192)	576	conv2d_951[0][0]
activation_948 (Activation)	(None, 12, 12, 192)	0	batch_normalization_960[0][0]
activation_951 (Activation)	(None, 12, 12, 192)	0	batch_normalization_963[0][0]
block17_16_mixed (Concatenate)	(None, 12, 12, 384)	0	activation_948[0][0] activation_951[0][0]
block17_16_conv (Conv2D)	(None, 12, 12, 1088)	418880	block17_16_mixed[0][0]
block17_16 (Lambda)	(None, 12, 12, 1088)	0	block17_15_ac[0][0] block17_16_conv[0][0]
block17_16_ac (Activation)	(None, 12, 12, 1088)	0	block17_16[0][0]
conv2d_953 (Conv2D)	(None, 12, 12, 128)	139264	block17_16_ac[0][0]
batch_normalization_965 (BatchN	(None, 12, 12, 128)	384	conv2d_953[0][0]
activation_953 (Activation)	(None, 12, 12, 128)	0	batch_normalization_965[0][0]
conv2d_954 (Conv2D)	(None, 12, 12, 160)	143360	activation_953[0][0]
batch_normalization_966 (BatchN	(None, 12, 12, 160)	480	conv2d_954[0][0]
activation_954 (Activation)	(None, 12, 12, 160)	0	batch_normalization_966[0][0]

conv2d_952 (Conv2D)	(None, 12, 12, 192)	208896	block17_16_ac[0][0]
conv2d_955 (Conv2D)	(None, 12, 12, 192)	215040	activation_954[0][0]
batch_normalization_964 (BatchN	(None, 12, 12, 192)	576	conv2d_952[0][0]
batch_normalization_967 (BatchN	(None, 12, 12, 192)	576	conv2d_955[0][0]
activation_952 (Activation)	(None, 12, 12, 192)	0	batch_normalization_964[0][0]
activation_955 (Activation)	(None, 12, 12, 192)	0	batch_normalization_967[0][0]
block17_17_mixed (Concatenate)	(None, 12, 12, 384)	0	activation_952[0][0] activation_955[0][0]
block17_17_conv (Conv2D)	(None, 12, 12, 1088)	418880	block17_17_mixed[0][0]
block17_17 (Lambda)	(None, 12, 12, 1088)	0	block17_16_ac[0][0] block17_17_conv[0][0]
block17_17_ac (Activation)	(None, 12, 12, 1088)	0	block17_17[0][0]
conv2d_957 (Conv2D)	(None, 12, 12, 128)	139264	block17_17_ac[0][0]
batch_normalization_969 (BatchN	(None, 12, 12, 128)	384	conv2d_957[0][0]
activation_957 (Activation)	(None, 12, 12, 128)	0	batch_normalization_969[0][0]
conv2d_958 (Conv2D)	(None, 12, 12, 160)	143360	activation_957[0][0]
batch_normalization_970 (BatchN	(None, 12, 12, 160)	480	conv2d_958[0][0]
activation_958 (Activation)	(None, 12, 12, 160)	0	batch_normalization_970[0][0]
conv2d_956 (Conv2D)	(None, 12, 12, 192)	208896	block17_17_ac[0][0]
conv2d_959 (Conv2D)	(None, 12, 12, 192)	215040	activation_958[0][0]
batch_normalization_968 (BatchN	(None, 12, 12, 192)	576	conv2d_956[0][0]
batch_normalization_971 (BatchN	(None, 12, 12, 192)	576	conv2d_959[0][0]
activation_956 (Activation)	(None, 12, 12, 192)	0	batch_normalization_968[0][0]
activation_959 (Activation)	(None, 12, 12, 192)	0	batch_normalization_971[0][0]
block17_18_mixed (Concatenate)	(None, 12, 12, 384)	0	activation_956[0][0] activation_959[0][0]
block17_18_conv (Conv2D)	(None, 12, 12, 1088)	418880	block17_18_mixed[0][0]
block17_18 (Lambda)	(None, 12, 12, 1088)	0	block17_17_ac[0][0] block17_18_conv[0][0]
block17_18_ac (Activation)	(None, 12, 12, 1088)	0	block17_18[0][0]
conv2d_961 (Conv2D)	(None, 12, 12, 128)	139264	block17_18_ac[0][0]
batch_normalization_973 (BatchN	(None, 12, 12, 128)	384	conv2d_961[0][0]
activation_961 (Activation)	(None, 12, 12, 128)	0	batch_normalization_973[0][0]
conv2d_962 (Conv2D)	(None, 12, 12, 160)	143360	activation_961[0][0]
batch_normalization_974 (BatchN	(None, 12, 12, 160)	480	conv2d_962[0][0]
activation_962 (Activation)	(None, 12, 12, 160)	0	batch_normalization_974[0][0]
conv2d_960 (Conv2D)	(None, 12, 12, 192)	208896	block17_18_ac[0][0]
conv2d_963 (Conv2D)	(None, 12, 12, 192)	215040	activation_962[0][0]
batch_normalization_972 (BatchN	(None, 12, 12, 192)	576	conv2d_960[0][0]
batch_normalization_975 (BatchN	(None, 12, 12, 192)	576	conv2d_963[0][0]
activation_960 (Activation)	(None, 12, 12, 192)	0	batch_normalization_972[0][0]
activation_963 (Activation)	(None, 12, 12, 192)	0	batch_normalization_975[0][0]
block17_19_mixed (Concatenate)	(None, 12, 12, 384)	0	activation_960[0][0] activation_963[0][0]
block17_19_conv (Conv2D)	(None, 12, 12, 1088)	418880	block17_19_mixed[0][0]
block17_19 (Lambda)	(None, 12, 12, 1088)	0	block17_18_ac[0][0] block17_19_conv[0][0]

block17_19_ac (Activation)	(None, 12, 12, 1088)	0	block17_19[0][0]
conv2d_965 (Conv2D)	(None, 12, 12, 128)	139264	block17_19_ac[0][0]
batch_normalization_977 (BatchN	(None, 12, 12, 128)	384	conv2d_965[0][0]
activation_965 (Activation)	(None, 12, 12, 128)	0	batch_normalization_977[0][0]
conv2d_966 (Conv2D)	(None, 12, 12, 160)	143360	activation_965[0][0]
batch_normalization_978 (BatchN	(None, 12, 12, 160)	480	conv2d_966[0][0]
activation_966 (Activation)	(None, 12, 12, 160)	0	batch_normalization_978[0][0]
conv2d_964 (Conv2D)	(None, 12, 12, 192)	208896	block17_19_ac[0][0]
conv2d_967 (Conv2D)	(None, 12, 12, 192)	215040	activation_966[0][0]
batch_normalization_976 (BatchN	(None, 12, 12, 192)	576	conv2d_964[0][0]
batch_normalization_979 (BatchN	(None, 12, 12, 192)	576	conv2d_967[0][0]
activation_964 (Activation)	(None, 12, 12, 192)	0	batch_normalization_976[0][0]
activation_967 (Activation)	(None, 12, 12, 192)	0	batch_normalization_979[0][0]
block17_20_mixed (Concatenate)	(None, 12, 12, 384)	0	activation_964[0][0] activation_967[0][0]
block17_20_conv (Conv2D)	(None, 12, 12, 1088)	418880	block17_20_mixed[0][0]
block17_20 (Lambda)	(None, 12, 12, 1088)	0	block17_19_ac[0][0] block17_20_conv[0][0]
block17_20_ac (Activation)	(None, 12, 12, 1088)	0	block17_20[0][0]
conv2d_972 (Conv2D)	(None, 12, 12, 256)	278528	block17_20_ac[0][0]
batch_normalization_984 (BatchN	(None, 12, 12, 256)	768	conv2d_972[0][0]
activation_972 (Activation)	(None, 12, 12, 256)	0	batch_normalization_984[0][0]
conv2d_968 (Conv2D)	(None, 12, 12, 256)	278528	block17_20_ac[0][0]
conv2d_970 (Conv2D)	(None, 12, 12, 256)	278528	block17_20_ac[0][0]
conv2d_973 (Conv2D)	(None, 12, 12, 288)	663552	activation_972[0][0]
batch_normalization_980 (BatchN	(None, 12, 12, 256)	768	conv2d_968[0][0]
batch_normalization_982 (BatchN	(None, 12, 12, 256)	768	conv2d_970[0][0]
batch_normalization_985 (BatchN	(None, 12, 12, 288)	864	conv2d_973[0][0]
activation_968 (Activation)	(None, 12, 12, 256)	0	batch_normalization_980[0][0]
activation_970 (Activation)	(None, 12, 12, 256)	0	batch_normalization_982[0][0]
activation_973 (Activation)	(None, 12, 12, 288)	0	batch_normalization_985[0][0]
conv2d_969 (Conv2D)	(None, 5, 5, 384)	884736	activation_968[0][0]
conv2d_971 (Conv2D)	(None, 5, 5, 288)	663552	activation_970[0][0]
conv2d_974 (Conv2D)	(None, 5, 5, 320)	829440	activation_973[0][0]
batch_normalization_981 (BatchN	(None, 5, 5, 384)	1152	conv2d_969[0][0]
batch_normalization_983 (BatchN	(None, 5, 5, 288)	864	conv2d_971[0][0]
batch_normalization_986 (BatchN	(None, 5, 5, 320)	960	conv2d_974[0][0]
activation_969 (Activation)	(None, 5, 5, 384)	0	batch_normalization_981[0][0]
activation_971 (Activation)	(None, 5, 5, 288)	0	batch_normalization_983[0][0]
activation_974 (Activation)	(None, 5, 5, 320)	0	batch_normalization_986[0][0]
max_pooling2d_19 (MaxPooling2D)	(None, 5, 5, 1088)	0	block17_20_ac[0][0]
mixed_7a (Concatenate)	(None, 5, 5, 2080)	0	activation_969[0][0] activation_971[0][0] activation_974[0][0] max_pooling2d_19[0][0]
conv2d_976 (Conv2D)	(None, 5, 5, 192)	399360	mixed_7a[0][0]

batch_normalization_988 (BatchN	(None, 5, 5, 192)	576	conv2d_976[0][0]
activation_976 (Activation)	(None, 5, 5, 192)	0	batch_normalization_988[0][0]
conv2d_977 (Conv2D)	(None, 5, 5, 224)	129024	activation_976[0][0]
batch_normalization_989 (BatchN	(None, 5, 5, 224)	672	conv2d_977[0][0]
activation_977 (Activation)	(None, 5, 5, 224)	0	batch_normalization_989[0][0]
conv2d_975 (Conv2D)	(None, 5, 5, 192)	399360	mixed_7a[0][0]
conv2d_978 (Conv2D)	(None, 5, 5, 256)	172032	activation_977[0][0]
batch_normalization_987 (BatchN	(None, 5, 5, 192)	576	conv2d_975[0][0]
batch_normalization_990 (BatchN	(None, 5, 5, 256)	768	conv2d_978[0][0]
activation_975 (Activation)	(None, 5, 5, 192)	0	batch_normalization_987[0][0]
activation_978 (Activation)	(None, 5, 5, 256)	0	batch_normalization_990[0][0]
block8_1_mixed (Concatenate)	(None, 5, 5, 448)	0	activation_975[0][0] activation_978[0][0]
block8_1_conv (Conv2D)	(None, 5, 5, 2080)	933920	block8_1_mixed[0][0]
block8_1 (Lambda)	(None, 5, 5, 2080)	0	mixed_7a[0][0] block8_1_conv[0][0]
block8_1_ac (Activation)	(None, 5, 5, 2080)	0	block8_1[0][0]
conv2d_980 (Conv2D)	(None, 5, 5, 192)	399360	block8_1_ac[0][0]
batch_normalization_992 (BatchN	(None, 5, 5, 192)	576	conv2d_980[0][0]
activation_980 (Activation)	(None, 5, 5, 192)	0	batch_normalization_992[0][0]
conv2d_981 (Conv2D)	(None, 5, 5, 224)	129024	activation_980[0][0]
batch_normalization_993 (BatchN	(None, 5, 5, 224)	672	conv2d_981[0][0]
activation_981 (Activation)	(None, 5, 5, 224)	0	batch_normalization_993[0][0]
conv2d_979 (Conv2D)	(None, 5, 5, 192)	399360	block8_1_ac[0][0]
conv2d_982 (Conv2D)	(None, 5, 5, 256)	172032	activation_981[0][0]
batch_normalization_991 (BatchN	(None, 5, 5, 192)	576	conv2d_979[0][0]
batch_normalization_994 (BatchN	(None, 5, 5, 256)	768	conv2d_982[0][0]
activation_979 (Activation)	(None, 5, 5, 192)	0	batch_normalization_991[0][0]
activation_982 (Activation)	(None, 5, 5, 256)	0	batch_normalization_994[0][0]
block8_2_mixed (Concatenate)	(None, 5, 5, 448)	0	activation_979[0][0] activation_982[0][0]
block8_2_conv (Conv2D)	(None, 5, 5, 2080)	933920	block8_2_mixed[0][0]
block8_2 (Lambda)	(None, 5, 5, 2080)	0	block8_1_ac[0][0] block8_2_conv[0][0]
block8_2_ac (Activation)	(None, 5, 5, 2080)	0	block8_2[0][0]
conv2d_984 (Conv2D)	(None, 5, 5, 192)	399360	block8_2_ac[0][0]
batch_normalization_996 (BatchN	(None, 5, 5, 192)	576	conv2d_984[0][0]
activation_984 (Activation)	(None, 5, 5, 192)	0	batch_normalization_996[0][0]
conv2d_985 (Conv2D)	(None, 5, 5, 224)	129024	activation_984[0][0]
batch_normalization_997 (BatchN	(None, 5, 5, 224)	672	conv2d_985[0][0]
activation_985 (Activation)	(None, 5, 5, 224)	0	batch_normalization_997[0][0]
conv2d_983 (Conv2D)	(None, 5, 5, 192)	399360	block8_2_ac[0][0]
conv2d_986 (Conv2D)	(None, 5, 5, 256)	172032	activation_985[0][0]
batch_normalization_995 (BatchN	(None, 5, 5, 192)	576	conv2d_983[0][0]
batch_normalization_998 (BatchN	(None, 5, 5, 256)	768	conv2d_986[0][0]
activation_983 (Activation)	(None, 5, 5, 192)	0	batch_normalization_995[0][0]

activation_986 (Activation)	(None, 5, 5, 256)	0	batch_normalization_998[0][0]
block8_3_mixed (Concatenate)	(None, 5, 5, 448)	0	activation_983[0][0] activation_986[0][0]
block8_3_conv (Conv2D)	(None, 5, 5, 2080)	933920	block8_3_mixed[0][0]
block8_3 (Lambda)	(None, 5, 5, 2080)	0	block8_2_ac[0][0] block8_3_conv[0][0]
block8_3_ac (Activation)	(None, 5, 5, 2080)	0	block8_3[0][0]
conv2d_988 (Conv2D)	(None, 5, 5, 192)	399360	block8_3_ac[0][0]
batch_normalization_1000 (Batch	(None, 5, 5, 192)	576	conv2d_988[0][0]
activation_988 (Activation)	(None, 5, 5, 192)	0	batch_normalization_1000[0][0]
conv2d_989 (Conv2D)	(None, 5, 5, 224)	129024	activation_988[0][0]
batch_normalization_1001 (Batch	(None, 5, 5, 224)	672	conv2d_989[0][0]
activation_989 (Activation)	(None, 5, 5, 224)	0	batch_normalization_1001[0][0]
conv2d_987 (Conv2D)	(None, 5, 5, 192)	399360	block8_3_ac[0][0]
conv2d_990 (Conv2D)	(None, 5, 5, 256)	172032	activation_989[0][0]
batch_normalization_999 (BatchN	(None, 5, 5, 192)	576	conv2d_987[0][0]
batch_normalization_1002 (Batch	(None, 5, 5, 256)	768	conv2d_990[0][0]
activation_987 (Activation)	(None, 5, 5, 192)	0	batch_normalization_999[0][0]
activation_990 (Activation)	(None, 5, 5, 256)	0	batch_normalization_1002[0][0]
block8_4_mixed (Concatenate)	(None, 5, 5, 448)	0	activation_987[0][0] activation_990[0][0]
block8_4_conv (Conv2D)	(None, 5, 5, 2080)	933920	block8_4_mixed[0][0]
block8_4 (Lambda)	(None, 5, 5, 2080)	0	block8_3_ac[0][0] block8_4_conv[0][0]
block8_4_ac (Activation)	(None, 5, 5, 2080)	0	block8_4[0][0]
conv2d_992 (Conv2D)	(None, 5, 5, 192)	399360	block8_4_ac[0][0]
batch_normalization_1004 (Batch	(None, 5, 5, 192)	576	conv2d_992[0][0]
activation_992 (Activation)	(None, 5, 5, 192)	0	batch_normalization_1004[0][0]
conv2d_993 (Conv2D)	(None, 5, 5, 224)	129024	activation_992[0][0]
batch_normalization_1005 (Batch	(None, 5, 5, 224)	672	conv2d_993[0][0]
activation_993 (Activation)	(None, 5, 5, 224)	0	batch_normalization_1005[0][0]
conv2d_991 (Conv2D)	(None, 5, 5, 192)	399360	block8_4_ac[0][0]
conv2d_994 (Conv2D)	(None, 5, 5, 256)	172032	activation_993[0][0]
batch_normalization_1003 (Batch	(None, 5, 5, 192)	576	conv2d_991[0][0]
batch_normalization_1006 (Batch	(None, 5, 5, 256)	768	conv2d_994[0][0]
activation_991 (Activation)	(None, 5, 5, 192)	0	batch_normalization_1003[0][0]
activation_994 (Activation)	(None, 5, 5, 256)	0	batch_normalization_1006[0][0]
block8_5_mixed (Concatenate)	(None, 5, 5, 448)	0	activation_991[0][0] activation_994[0][0]
block8_5_conv (Conv2D)	(None, 5, 5, 2080)	933920	block8_5_mixed[0][0]
block8_5 (Lambda)	(None, 5, 5, 2080)	0	block8_4_ac[0][0] block8_5_conv[0][0]
block8_5_ac (Activation)	(None, 5, 5, 2080)	0	block8_5[0][0]
conv2d_996 (Conv2D)	(None, 5, 5, 192)	399360	block8_5_ac[0][0]
batch_normalization_1008 (Batch	(None, 5, 5, 192)	576	conv2d_996[0][0]
activation_996 (Activation)	(None, 5, 5, 192)	0	batch_normalization_1008[0][0]
conv2d_997 (Conv2D)	(None, 5, 5, 224)	129024	activation_996[0][0]

batch_normalization_1009 (Batch	(None, 5, 5, 224)	672	conv2d_997[0][0]
activation_997 (Activation)	(None, 5, 5, 224)	0	batch_normalization_1009[0][0]
conv2d_995 (Conv2D)	(None, 5, 5, 192)	399360	block8_5_ac[0][0]
conv2d_998 (Conv2D)	(None, 5, 5, 256)	172032	activation_997[0][0]
batch_normalization_1007 (Batch	(None, 5, 5, 192)	576	conv2d_995[0][0]
batch_normalization_1010 (Batch	(None, 5, 5, 256)	768	conv2d_998[0][0]
activation_995 (Activation)	(None, 5, 5, 192)	0	batch_normalization_1007[0][0]
activation_998 (Activation)	(None, 5, 5, 256)	0	batch_normalization_1010[0][0]
block8_6_mixed (Concatenate)	(None, 5, 5, 448)	0	activation_995[0][0] activation_998[0][0]
block8_6_conv (Conv2D)	(None, 5, 5, 2080)	933920	block8_6_mixed[0][0]
block8_6 (Lambda)	(None, 5, 5, 2080)	0	block8_5_ac[0][0] block8_6_conv[0][0]
block8_6_ac (Activation)	(None, 5, 5, 2080)	0	block8_6[0][0]
conv2d_1000 (Conv2D)	(None, 5, 5, 192)	399360	block8_6_ac[0][0]
batch_normalization_1012 (Batch	(None, 5, 5, 192)	576	conv2d_1000[0][0]
activation_1000 (Activation)	(None, 5, 5, 192)	0	batch_normalization_1012[0][0]
conv2d_1001 (Conv2D)	(None, 5, 5, 224)	129024	activation_1000[0][0]
batch_normalization_1013 (Batch	(None, 5, 5, 224)	672	conv2d_1001[0][0]
activation_1001 (Activation)	(None, 5, 5, 224)	0	batch_normalization_1013[0][0]
conv2d_999 (Conv2D)	(None, 5, 5, 192)	399360	block8_6_ac[0][0]
conv2d_1002 (Conv2D)	(None, 5, 5, 256)	172032	activation_1001[0][0]
batch_normalization_1011 (Batch	(None, 5, 5, 192)	576	conv2d_999[0][0]
batch_normalization_1014 (Batch	(None, 5, 5, 256)	768	conv2d_1002[0][0]
activation_999 (Activation)	(None, 5, 5, 192)	0	batch_normalization_1011[0][0]
activation_1002 (Activation)	(None, 5, 5, 256)	0	batch_normalization_1014[0][0]
block8_7_mixed (Concatenate)	(None, 5, 5, 448)	0	activation_999[0][0] activation_1002[0][0]
block8_7_conv (Conv2D)	(None, 5, 5, 2080)	933920	block8_7_mixed[0][0]
block8_7 (Lambda)	(None, 5, 5, 2080)	0	block8_6_ac[0][0] block8_7_conv[0][0]
block8_7_ac (Activation)	(None, 5, 5, 2080)	0	block8_7[0][0]
conv2d_1004 (Conv2D)	(None, 5, 5, 192)	399360	block8_7_ac[0][0]
batch_normalization_1016 (Batch	(None, 5, 5, 192)	576	conv2d_1004[0][0]
activation_1004 (Activation)	(None, 5, 5, 192)	0	batch_normalization_1016[0][0]
conv2d_1005 (Conv2D)	(None, 5, 5, 224)	129024	activation_1004[0][0]
batch_normalization_1017 (Batch	(None, 5, 5, 224)	672	conv2d_1005[0][0]
activation_1005 (Activation)	(None, 5, 5, 224)	0	batch_normalization_1017[0][0]
conv2d_1003 (Conv2D)	(None, 5, 5, 192)	399360	block8_7_ac[0][0]
conv2d_1006 (Conv2D)	(None, 5, 5, 256)	172032	activation_1005[0][0]
batch_normalization_1015 (Batch	(None, 5, 5, 192)	576	conv2d_1003[0][0]
batch_normalization_1018 (Batch	(None, 5, 5, 256)	768	conv2d_1006[0][0]
activation_1003 (Activation)	(None, 5, 5, 192)	0	batch_normalization_1015[0][0]
activation_1006 (Activation)	(None, 5, 5, 256)	0	batch_normalization_1018[0][0]
block8_8_mixed (Concatenate)	(None, 5, 5, 448)	0	activation_1003[0][0] activation_1006[0][0]
block8_8_conv (Conv2D)	(None, 5, 5, 2080)	933920	block8_8_mixed[0][0]

block8_8 (Lambda)	(None, 5, 5, 2080)	0	block8_7_ac[0][0] block8_8_conv[0][0]
block8_8_ac (Activation)	(None, 5, 5, 2080)	0	block8_8[0][0]
conv2d_1008 (Conv2D)	(None, 5, 5, 192)	399360	block8_8_ac[0][0]
batch_normalization_1020 (Batch	(None, 5, 5, 192)	576	conv2d_1008[0][0]
activation_1008 (Activation)	(None, 5, 5, 192)	0	batch_normalization_1020[0][0]
conv2d_1009 (Conv2D)	(None, 5, 5, 224)	129024	activation_1008[0][0]
batch_normalization_1021 (Batch	(None, 5, 5, 224)	672	conv2d_1009[0][0]
activation_1009 (Activation)	(None, 5, 5, 224)	0	batch_normalization_1021[0][0]
conv2d_1007 (Conv2D)	(None, 5, 5, 192)	399360	block8_8_ac[0][0]
conv2d_1010 (Conv2D)	(None, 5, 5, 256)	172032	activation_1009[0][0]
batch_normalization_1019 (Batch	(None, 5, 5, 192)	576	conv2d_1007[0][0]
batch_normalization_1022 (Batch	(None, 5, 5, 256)	768	conv2d_1010[0][0]
activation_1007 (Activation)	(None, 5, 5, 192)	0	batch_normalization_1019[0][0]
activation_1010 (Activation)	(None, 5, 5, 256)	0	batch_normalization_1022[0][0]
block8_9_mixed (Concatenate)	(None, 5, 5, 448)	0	activation_1007[0][0] activation_1010[0][0]
block8_9_conv (Conv2D)	(None, 5, 5, 2080)	933920	block8_9_mixed[0][0]
block8_9 (Lambda)	(None, 5, 5, 2080)	0	block8_8_ac[0][0] block8_9_conv[0][0]
block8_9_ac (Activation)	(None, 5, 5, 2080)	0	block8_9[0][0]
conv2d_1012 (Conv2D)	(None, 5, 5, 192)	399360	block8_9_ac[0][0]
batch_normalization_1024 (Batch	(None, 5, 5, 192)	576	conv2d_1012[0][0]
activation_1012 (Activation)	(None, 5, 5, 192)	0	batch_normalization_1024[0][0]
conv2d_1013 (Conv2D)	(None, 5, 5, 224)	129024	activation_1012[0][0]
batch_normalization_1025 (Batch	(None, 5, 5, 224)	672	conv2d_1013[0][0]
activation_1013 (Activation)	(None, 5, 5, 224)	0	batch_normalization_1025[0][0]
conv2d_1011 (Conv2D)	(None, 5, 5, 192)	399360	block8_9_ac[0][0]
conv2d_1014 (Conv2D)	(None, 5, 5, 256)	172032	activation_1013[0][0]
batch_normalization_1023 (Batch	(None, 5, 5, 192)	576	conv2d_1011[0][0]
batch_normalization_1026 (Batch	(None, 5, 5, 256)	768	conv2d_1014[0][0]
activation_1011 (Activation)	(None, 5, 5, 192)	0	batch_normalization_1023[0][0]
activation_1014 (Activation)	(None, 5, 5, 256)	0	batch_normalization_1026[0][0]
block8_10_mixed (Concatenate)	(None, 5, 5, 448)	0	activation_1011[0][0] activation_1014[0][0]
block8_10_conv (Conv2D)	(None, 5, 5, 2080)	933920	block8_10_mixed[0][0]
block8_10 (Lambda)	(None, 5, 5, 2080)	0	block8_9_ac[0][0] block8_10_conv[0][0]
conv_7b (Conv2D)	(None, 5, 5, 1536)	3194880	block8_10[0][0]
conv_7b_bn (BatchNormalization)	(None, 5, 5, 1536)	4608	conv_7b[0][0]
conv_7b_ac (Activation)	(None, 5, 5, 1536)	0	conv_7b_bn[0][0]
reshape_4 (Reshape)	(None, 25, 1536)	0	conv_7b_ac[0][0]
lstm_4 (LSTM)	(None, 25, 512)	4196352	reshape_4[0][0]
batch_normalization_1027 (Batch	(None, 25, 512)	2048	lstm_4[0][0]
flatten (Flatten)	(None, 12800)	0	batch_normalization_1027[0][0]
dense_12 (Dense)	(None, 4096)	52432896	flatten[0][0]

```
batch_normalization_1028 (Batch (None, 4096)         16384       dense_12[0][0]
_____________________________________________________________________________________
dense_13 (Dense)                (None, 4096)         16781312    batch_normalization_1028[0][0]
_____________________________________________________________________________________
batch_normalization_1029 (Batch (None, 4096)         16384       dense_13[0][0]
_____________________________________________________________________________________
dense_14 (Dense)                (None, 4)            16388       batch_normalization_1029[0][0]
=====================================================================================
Total params: 127,798,500
Trainable params: 69,248,004
Non-trainable params: 58,550,496
_____________________________________________________________________________________
```