EMERGING METHODS FOR EARLY DETECTION OF FORESTFIRES

MODEL

BUILDINGSAVE

THEMODEL

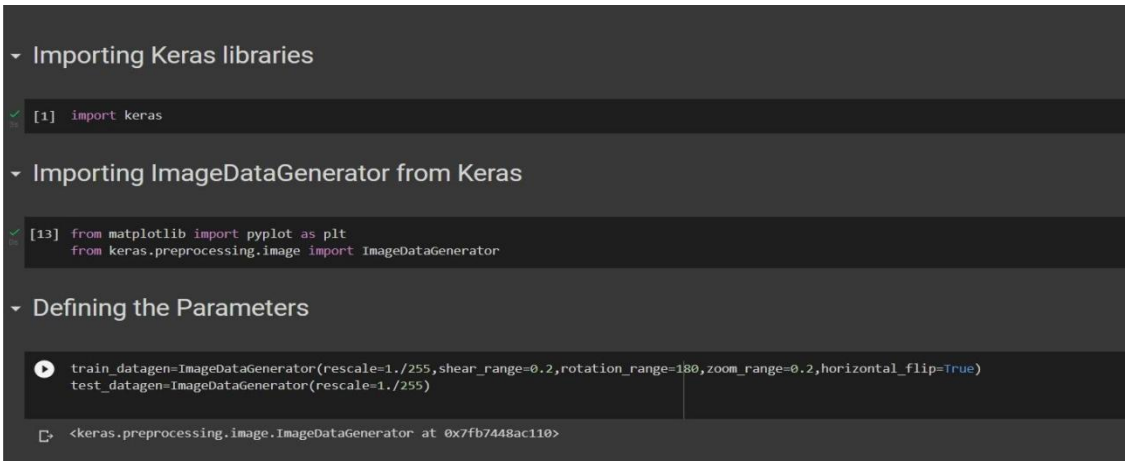
TeamID	PNT2022TMID30362
ProjectName	Project-Emerging Methods For Early detection of forest fires

SAVETHEMODEL

Yourmodelistobesavedforfuturepurposes.Thissavedmodelalsoisintegratedwithanandroidapplication orweb application inorder to predict something.

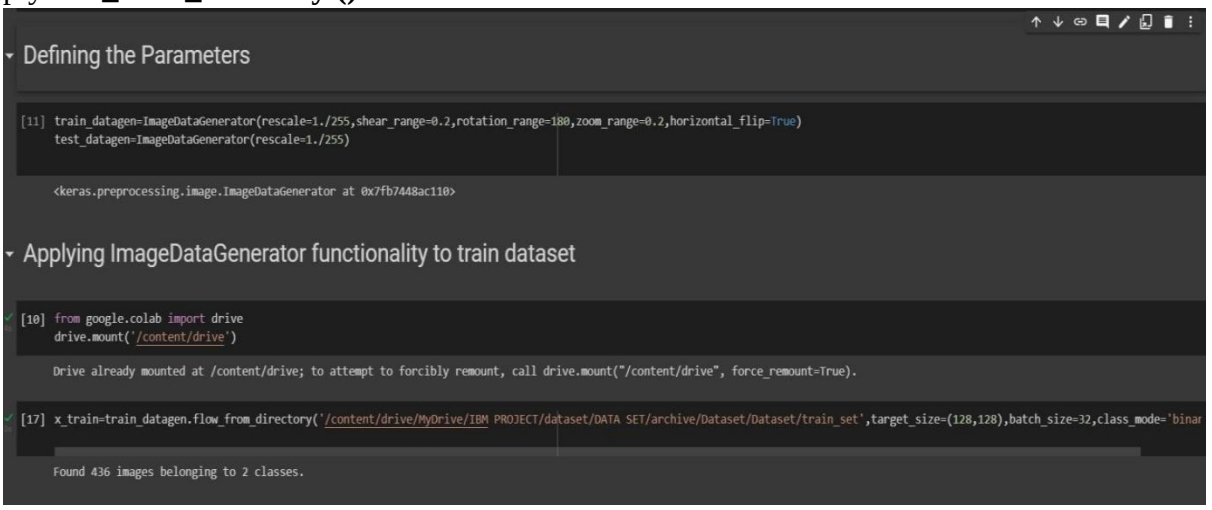
IMPORTLIBRARIES:

IMPORTImageDataGenerator FROMKERAS:



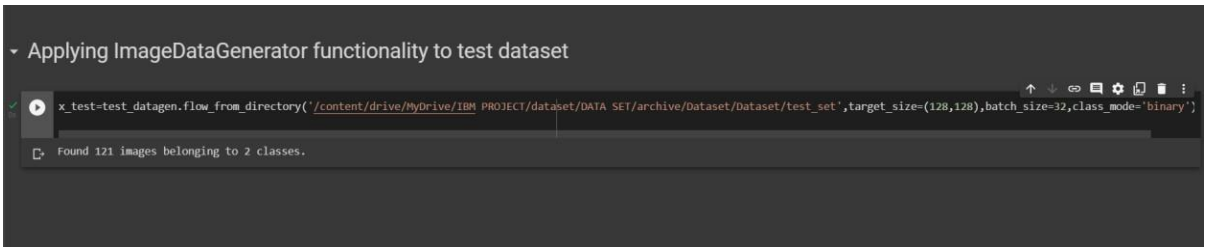
APPLYINGImageDataGeneratortotraindataset:

ply**flow_from_directory ()**methodforTrainfolder.



APPLYINGImageDataGeneratortotestdataset:

Applyingthe**flow_from_directory()**methodfortestfolder.



IMPORTINGMODELBUILDINGLIBRARIES:

11/8/22, 1:16 AM Main code - Colaboratory

Importing Model Building Libraries

```
#to define the linear Initialisation import sequential
from keras.models import Sequential
#to add layers import Dense
from keras.layers import Dense
#to create Convolutional kernel import convolution2D
from keras.layers import Convolution2D
#import Maxpooling layer
from keras.layers import MaxPooling2D
#import flatten layer
from keras.layers import Flatten
import warnings
warnings.filterwarnings('ignore')
```

INITIALIZINGTHEMODEL:

▾ Initializing the model

```
model=Sequential()
```

ADDINGCNNLAYERS:

▾ Adding CNN Layers

```
model.add(Convolution2D(32,(3,3),input_shape=(128,128,3),activation='relu'))
#add maxpooling layers
model.add(MaxPooling2D(pool_size=(2,2)))
#add faltten layer
model.add(Flatten())
```

ADDINGDENSELAYERS:

▾ Add Dense layers

```
#add hidden layers
model.add(Dense(150,activation='relu'))
#add output layer
model.add(Dense(1,activation='sigmoid'))
```

CONFIGURINGTHELEARNING PROCESS:

▾ configuring the learning process

```
model.compile(loss='binary_crossentropy',optimizer="adam",metrics=["accuracy"])
```

TRAININGTHEMODEL:

▾ Training the model

```
model.fit_generator(x_train,steps_per_epoch=14,epochs=10,validation_data=x_test,validation
```

```
Epoch 1/10
14/14 [==============================] - 322s 19s/step - loss: 1.5998 - accuracy: 0.7
Epoch 2/10
14/14 [==============================] - 26s 2s/step - loss: 0.3427 - accuracy: 0.862
Epoch 3/10
14/14 [==============================] - 32s 2s/step - loss: 0.2979 - accuracy: 0.885
Epoch 4/10
14/14 [==============================] - 29s 2s/step - loss: 0.2585 - accuracy: 0.897
Epoch 5/10
14/14 [==============================] - 29s 2s/step - loss: 0.1926 - accuracy: 0.924
Epoch 6/10
14/14 [==============================] - 30s 2s/step - loss: 0.1971 - accuracy: 0.926
Epoch 7/10
14/14 [==============================] - 32s 2s/step - loss: 0.1781 - accuracy: 0.928
Epoch 8/10
14/14 [==============================] - 30s 2s/step - loss: 0.1796 - accuracy: 0.924
Epoch 9/10
14/14 [==============================] - 31s 2s/step - loss: 0.2306 - accuracy: 0.896
Epoch 10/10
14/14 [==============================] - 27s 2s/step - loss: 0.2593 - accuracy: 0.889
<keras.callbacks.History at 0x7fd537101390>
```

SAVETHE MODEL:

▾ Save the model

```
model.save("forest.h5")
```