

ASSIGNMENT - 3  
C SRINATH  
20MID0023  
VIT VELLORE CAMPUS

In [ ]: *#Loding the train data and test data*

In [2]: *#!/content/drive/MyDrive/datasets/bird\_dataset/test\_data.zip*  
*!unzip '/content/drive/MyDrive/datasets/bird\_dataset/test\_data.zip'*

```
Archive: /content/drive/MyDrive/datasets/bird_dataset/test_data.zip
  creating: test_data/test_data/
  creating: test_data/test_data/blasti/
 inflating: test_data/test_data/blasti/DSC_6396.jpg
 inflating: test_data/test_data/blasti/DSC_6397.jpg
 inflating: test_data/test_data/blasti/DSC_6398.jpg
 inflating: test_data/test_data/blasti/DSC_6399.jpg
 inflating: test_data/test_data/blasti/DSC_6400.jpg
 inflating: test_data/test_data/blasti/DSC_6401.jpg
 inflating: test_data/test_data/blasti/DSC_6402.jpg
 inflating: test_data/test_data/blasti/DSC_6403.jpg
 inflating: test_data/test_data/blasti/DSC_6405.jpg
 inflating: test_data/test_data/blasti/DSC_6406.jpg
 inflating: test_data/test_data/blasti/DSC_6407.jpg
 inflating: test_data/test_data/blasti/DSC_6408.jpg
 inflating: test_data/test_data/blasti/DSC_6409.jpg
 inflating: test_data/test_data/blasti/DSC_6410.jpg
 inflating: test_data/test_data/blasti/DSC_6411.jpg
  creating: test_data/test_data/bonegl/
 inflating: test_data/test_data/bonegl/DSC_4507.jpg
```

```
In [3]: #/content/drive/MyDrive/datasets/bird_dataset/train_data.zip
!unzip '/content/drive/MyDrive/datasets/bird_dataset/train_data.zip'
```

```
Archive:  /content/drive/MyDrive/datasets/bird_dataset/train_data.zip
  creating: train_data/train_data/
  creating: train_data/train_data/blasti/
inflating: train_data/train_data/blasti/DSC_6382.jpg
inflating: train_data/train_data/blasti/DSC_6383.jpg
inflating: train_data/train_data/blasti/DSC_6384.jpg
inflating: train_data/train_data/blasti/DSC_6384-2.jpg
inflating: train_data/train_data/blasti/DSC_6385.jpg
inflating: train_data/train_data/blasti/DSC_6386.jpg
inflating: train_data/train_data/blasti/DSC_6387.jpg
inflating: train_data/train_data/blasti/DSC_6388.jpg
inflating: train_data/train_data/blasti/DSC_6389.jpg
inflating: train_data/train_data/blasti/DSC_6390.jpg
inflating: train_data/train_data/blasti/DSC_6391.jpg
inflating: train_data/train_data/blasti/DSC_6392.jpg
inflating: train_data/train_data/blasti/DSC_6393.jpg
inflating: train_data/train_data/blasti/DSC_6394.jpg
inflating: train_data/train_data/blasti/DSC_6395.jpg
  creating: train_data/train_data/bonegl/
  inflating: train_data/train_data/bonegl/DSC_6396.jpg
```

```
In [4]: #Data augmentation step
from tensorflow.keras.preprocessing.image import ImageDataGenerator
```

```
In [5]: train_gen = ImageDataGenerator(rescale=(1./255),horizontal_flip=True,shear_range=0.3)
```

```
In [6]: #Convert img to 0 and 1 (binary)
test_gen=ImageDataGenerator(rescale=(1./255))
```

```
In [7]: train = train_gen.flow_from_directory ('/content/train_data/train_data',
                                              target_size=(120,120),
                                              class_mode='categorical',
                                              batch_size=8)
```

Found 150 images belonging to 16 classes.

```
In [8]: test = test_gen.flow_from_directory ('/content/test_data/test_data',  
                                             target_size=(120,120),  
                                             class_mode='categorical',  
                                             batch_size=8)
```

Found 157 images belonging to 16 classes.

```
In [9]: #Building CNN model  
from tensorflow.keras.layers import Convolution2D,MaxPooling2D,Flatten,Dense  
from tensorflow.keras.models import Sequential  
  
model=Sequential()  
model.add(Convolution2D(10,(3,3),activation='relu',input_shape=(120,120,3)))  
model.add(MaxPooling2D(pool_size=(2,2)))  
model.add(Flatten())  
model.add(Dense(45,activation='relu'))  
model.add(Dense(4,activation='softmax'))
```

```
In [10]: #Model tuning  
model=Sequential()  
model.add(Convolution2D(10,(3,3),activation='relu',input_shape=(120,120,3)))  
model.add(MaxPooling2D(pool_size=(2,2)))  
model.add(Convolution2D(20,(3,3),activation='relu',))  
model.add(MaxPooling2D(pool_size=(2,2)))  
model.add(Convolution2D(30,(3,3),activation='relu',))  
model.add(MaxPooling2D(pool_size=(2,2)))  
model.add(Flatten())  
model.add(Dense(32,activation='relu'))  
model.add(Dense(64,activation='relu'))  
model.add(Dense(16,activation='softmax'))
```

```
In [11]: train.class_indices
```

```
Out[11]: {'blasti': 0,  
          'bonegl': 1,  
          'brhkyt': 2,  
          'cbrtsh': 3,  
          'cmnmyn': 4,  
          'gretit': 5,  
          'hilpig': 6,  
          'himbul': 7,  
          'himgri': 8,  
          'hsparo': 9,  
          'indvul': 10,  
          'jglowl': 11,  
          'lbicrw': 12,  
          'mgprob': 13,  
          'rebing': 14,  
          'wcrsrt': 15}
```

```
In [12]: model.compile(optimizer='adam',loss='categorical_crossentropy',metrics=['accuracy'])
```

```
In [13]: model.fit(train,batch_size=8,validation_data=test,epochs=10)
```

```
Epoch 1/10
19/19 [=====] - 122s 7s/step - loss: 2.7523 - accuracy: 0.0800 - val_loss: 2.7225 - val_accu
racy: 0.1210
Epoch 2/10
19/19 [=====] - 128s 7s/step - loss: 2.6848 - accuracy: 0.1533 - val_loss: 2.6492 - val_accu
racy: 0.2102
Epoch 3/10
19/19 [=====] - 127s 7s/step - loss: 2.5385 - accuracy: 0.1800 - val_loss: 2.7074 - val_accu
racy: 0.1146
Epoch 4/10
19/19 [=====] - 127s 7s/step - loss: 2.3770 - accuracy: 0.2267 - val_loss: 2.6269 - val_accu
racy: 0.2420
Epoch 5/10
19/19 [=====] - 130s 7s/step - loss: 2.2220 - accuracy: 0.2800 - val_loss: 2.5929 - val_accu
racy: 0.2484
Epoch 6/10
19/19 [=====] - 111s 6s/step - loss: 2.1557 - accuracy: 0.2600 - val_loss: 2.6686 - val_accu
racy: 0.2102
Epoch 7/10
19/19 [=====] - 129s 7s/step - loss: 2.0189 - accuracy: 0.3467 - val_loss: 2.7427 - val_accu
racy: 0.2739
Epoch 8/10
19/19 [=====] - 127s 7s/step - loss: 1.9127 - accuracy: 0.3067 - val_loss: 2.7145 - val_accu
racy: 0.2548
Epoch 9/10
19/19 [=====] - 129s 7s/step - loss: 1.7562 - accuracy: 0.3733 - val_loss: 2.9237 - val_accu
racy: 0.2611
Epoch 10/10
19/19 [=====] - 111s 6s/step - loss: 1.6578 - accuracy: 0.4467 - val_loss: 2.9407 - val_accu
racy: 0.1847
```

```
Out[13]: <keras.callbacks.History at 0x7f569c2fd8d0>
```

```
In [14]: model.save('animalWeights.h5')
```

Testing the model

```
In [15]: import numpy as np
from tensorflow.keras.preprocessing import image
```

```
In [16]: #Testing1
# /content/himgri_egyptian_vulture_1.jfif
img1 = image.load_img('/content/himgri_egyptian_vulture_1.jfif',target_size=(120,120))
print(img1)
img1 = image.img_to_array(img1)
img1 = np.expand_dims(img1,axis=0)
ypred=np.argmax(model.predict(img1))
pred=['blasti','bonegl','brhkyt','cbrtsh','cmnmyn','gretit','hilpig','himbul','himgri','hsparo','indvul','jglowl','lbi']
print(pred[ypred])
```

<PIL.Image.Image image mode=RGB size=120x120 at 0x7F569F62FF10>  
1/1 [=====] - 0s 169ms/step  
himgri

```
In [17]: #Testing2
# /content/himgri_egyptian_vulture_2.jfif
img2 = image.load_img('/content/himgri_egyptian_vulture_2.jfif',target_size=(120,120))
print(img2)
img2 = image.img_to_array(img2)
img2 = np.expand_dims(img2,axis=0)
ypred=np.argmax(model.predict(img2))
pred=['blasti','bonegl','brhkyt','cbrtsh','cmnmyn','gretit','hilpig','himbul','himgri','hsparo','indvul','jglowl','lbi']
print(pred[ypred])
```

<PIL.Image.Image image mode=RGB size=120x120 at 0x7F568F6A2DD0>  
1/1 [=====] - 0s 27ms/step  
blasti

```
In [18]: #Testing3
# /content/wcsrt_white-capped redstart_1.jfif
img3 = image.load_img('/content/wcsrt_white-capped redstart_1.jfif',target_size=(120,120))
print(img3)
img3 = image.img_to_array(img3)
img3 = np.expand_dims(img3,axis=0)
ypred=np.argmax(model.predict(img3))
pred=['blasti','bonegl','brhkyt','cbrtsh','cmnmyn','gretit','hilpig','himbul','himgri','hsparo','indvul','jglowl','lbi
print(pred[ypred])
```

<PIL.Image.Image image mode=RGB size=120x120 at 0x7F568921EF20>  
1/1 [=====] - 0s 41ms/step  
blasti

```
In [19]: #Testing4
# /content/wcsrt_white-capped redstart_2.jfif
img4 = image.load_img('/content/wcsrt_white-capped redstart_2.jfif',target_size=(120,120))
print(img4)
img4 = image.img_to_array(img4)
img4 = np.expand_dims(img4,axis=0)
ypred=np.argmax(model.predict(img4))
pred=['blasti','bonegl','brhkyt','cbrtsh','cmnmyn','gretit','hilpig','himbul','himgri','hsparo','indvul','jglowl','lbi
print(pred[ypred])
```

<PIL.Image.Image image mode=RGB size=120x120 at 0x7F5729C62F20>  
1/1 [=====] - 0s 27ms/step  
bonegl