ASSIGNMENT 3

ASSIGNMENT DATE	1 OCTOBER 2022
STUDENT NAME	HARI @ ARJUN.K
STUDENT ROLL NUMBER	2019504525
MAXIMUM MARKS	2 MARKS

Importing Libraries

```
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Dense
from tensorflow.keras.layers import Convolution2D
from tensorflow.keras.layers import MaxPooling2D
from tensorflow.keras.layers import Flatten
from sklearn.model_selection import train_test_split

from tensorflow.keras.preprocessing.image import ImageDataGenerator
```

Image Augmentation

```
train_datagen=ImageDataGenerator(rescale=1./255,shear_range=0.2,zoom_r
ange=0.2,horizontal_flip=True,vertical_flip=True)

test_datagen=ImageDataGenerator(rescale=1./255)

x_train = train_datagen.flow_from_directory(r"C:\Users\Karthikeyan\
Downloads\Flowers-Dataset\flowers\
train",target_size=(128,128),batch_size=32,class_mode="categorical")

Found 3451 images belonging to 5 classes.

x_test=x_test = test_datagen.flow_from_directory(r"C:\Users\
Karthikeyan\Downloads\Flowers-Dataset\flowers\
test",target_size=(128,128),batch_size=32,class_mode="categorical")

Found 866 images belonging to 5 classes.

x_train.class_indices

{'daisy': 0, 'dandelion': 1, 'rose': 2, 'sunflower': 3, 'tulip': 4}
```

Creating Model

```
model = Sequential()
```

Adding Layers

```
model.add(Convolution2D(32,
(3,3),input_shape=(128,128,3),activation='relu'))

model.add(MaxPooling2D(pool_size=(2,2)))

model.add(Flatten())

model.add(Dense(units=200,kernel_initializer="random_uniform",activati
on="relu"))
```

```
model.add(Dense(units=300,kernel_initializer="random_uniform",activati
on="relu"))

model.add(Dense(units=400,kernel_initializer="random_uniform",activati
on="relu"))

model.add(Dense(units=500,kernel_initializer="random_uniform",activati
on="relu"))

model.add(Dense(units=5,kernel_initializer="random_uniform",activation
="softmax"))
```

Compiling Model

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

Fitting the Model

```
model.fit_generator(x_train,steps_per_epoch=39,epochs=25,validation_da
ta=x_test,validation_steps=10)

C:\Users\Karthikeyan\AppData\Local\Temp\
ipykernel_14112\3505885595.py:1: UserWarning: `Model.fit_generator` is
deprecated and will be removed in a future version. Please use
`Model.fit`, which supports generators.

model.fit_generator(x_train,steps_per_epoch=39,epochs=25,validation_da
ta=x_test,validation_steps=10)

Epoch 1/25
39/39 [==============================] - 50s 1s/step - loss: 1.5990 -
accuracy: 0.2486 - val_loss: 1.5958 - val_accuracy: 0.2594
Epoch 2/25
39/39 [==============================] - 36s 914ms/step - loss: 1.5964
- accuracy: 0.2476 - val_loss: 1.6038 - val_accuracy: 0.2219
Epoch 3/25
39/39 [==============================] - 36s 913ms/step - loss: 1.5932
- accuracy: 0.2500 - val_loss: 1.6053 - val_accuracy: 0.2188
Epoch 4/25
39/39 [==============================] - 35s 878ms/step - loss: 1.5996
- accuracy: 0.2179 - val_loss: 1.5977 - val_accuracy: 0.2438
Epoch 5/25
39/39 [==============================] - 39s 993ms/step - loss: 1.5987
- accuracy: 0.2542 - val_loss: 1.6019 - val_accuracy: 0.2156
Epoch 6/25
39/39 [==============================] - 70s 2s/step - loss: 1.5955 -
accuracy: 0.2446 - val_loss: 1.6018 - val_accuracy: 0.2531
Epoch 7/25
39/39 [==============================] - 43s 1s/step - loss: 1.6017 -
```

```
accuracy: 0.2420 - val_loss: 1.6017 - val_accuracy: 0.2313
Epoch 8/25
39/39 [==============================] - 78s 2s/step - loss: 1.5978 -
accuracy: 0.2468 - val_loss: 1.6131 - val_accuracy: 0.2219
Epoch 9/25
39/39 [==============================] - 40s 1s/step - loss: 1.5996 -
accuracy: 0.2414 - val_loss: 1.5982 - val_accuracy: 0.2375
Epoch 10/25
39/39 [==============================] - 71s 2s/step - loss: 1.5980 -
accuracy: 0.2324 - val_loss: 1.5958 - val_accuracy: 0.2781
Epoch 11/25
39/39 [==============================] - 38s 973ms/step - loss: 1.6007
- accuracy: 0.2236 - val_loss: 1.5995 - val_accuracy: 0.2219
Epoch 12/25
39/39 [==============================] - 76s 2s/step - loss: 1.6020 -
accuracy: 0.2444 - val_loss: 1.5980 - val_accuracy: 0.2469
Epoch 13/25
39/39 [==============================] - 77s 2s/step - loss: 1.6007 -
accuracy: 0.2404 - val_loss: 1.5918 - val_accuracy: 0.2594
Epoch 14/25
39/39 [==============================] - 71s 2s/step - loss: 1.5983 -
accuracy: 0.2365 - val_loss: 1.5931 - val_accuracy: 0.2438
Epoch 15/25
39/39 [==============================] - 34s 862ms/step - loss: 1.6007
- accuracy: 0.2428 - val_loss: 1.6041 - val_accuracy: 0.2406
Epoch 16/25
39/39 [==============================] - 67s 2s/step - loss: 1.6055 -
accuracy: 0.2261 - val_loss: 1.5886 - val_accuracy: 0.2719
Epoch 17/25
39/39 [==============================] - 74s 2s/step - loss: 1.6003 -
accuracy: 0.2364 - val_loss: 1.5860 - val_accuracy: 0.2531
Epoch 18/25
39/39 [==============================] - 67s 2s/step - loss: 1.5970 -
accuracy: 0.2430 - val_loss: 1.6006 - val_accuracy: 0.2469
Epoch 19/25
39/39 [==============================] - 63s 2s/step - loss: 1.5953 -
accuracy: 0.2700 - val_loss: 1.5719 - val_accuracy: 0.3063
Epoch 20/25
39/39 [==============================] - 32s 815ms/step - loss: 1.6037
- accuracy: 0.2348 - val_loss: 1.5886 - val_accuracy: 0.2875
Epoch 21/25
39/39 [==============================] - 67s 2s/step - loss: 1.5935 -
accuracy: 0.2516 - val_loss: 1.6085 - val_accuracy: 0.2313
Epoch 22/25
39/39 [==============================] - 70s 2s/step - loss: 1.5959 -
accuracy: 0.2500 - val_loss: 1.5986 - val_accuracy: 0.2469
Epoch 23/25
39/39 [==============================] - 74s 2s/step - loss: 1.5973 -
accuracy: 0.2452 - val_loss: 1.5905 - val_accuracy: 0.2875
Epoch 24/25
```

```
39/39 [==============================] - 72s 2s/step - loss: 1.5973 -
accuracy: 0.2588 - val_loss: 1.6028 - val_accuracy: 0.2375
Epoch 25/25
39/39 [==============================] - 92s 2s/step - loss: 1.5974 -
accuracy: 0.2476 - val_loss: 1.6074 - val_accuracy: 0.2250

<keras.callbacks.History at 0x22198667d30>
```

Saving the Model

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

Testing the Model

```
from tensorflow.keras.models import load_model
from tensorflow.keras.preprocessing import image
import numpy as np

model=load_model("flowers.h5")

img=image.load_img(r"C:\Users\Karthikeyan\Downloads\Flowers-Dataset\
flowers\flower.jpg",target_size=(128,128))
img
```

```
x=image.img_to_array(img)
x

array([[[  2.,   2.,   0.],
        [  2.,   2.,   0.],
        [  2.,   2.,   0.],
        ...,
        [  2.,   1.,   0.],
        [  2.,   1.,   0.],
        [  2.,   1.,   0.]],

       [[  4.,   3.,   1.],
        [  4.,   3.,   1.],
        [  4.,   3.,   1.],
        ...,
        [  2.,   1.,   0.],
```

ASSIGNMENT 3

```
        [  2.,   1.,   0.],
        [  2.,   1.,   0.]],

       [[  5.,   1.,   0.],
        [  6.,   2.,   1.],
        [  6.,   2.,   1.],
        ...,
        [  2.,   1.,   0.],
        [  2.,   1.,   0.],
        [  2.,   1.,   0.]],

       ...,

       [[ 62., 109.,   3.],
        [ 72., 125.,   0.],
        [ 94., 147.,   3.],
        ...,
        [105., 160.,   6.],
        [ 91., 147.,   0.],
        [116., 176.,   0.]],

       [[ 88., 141.,   0.],
        [ 94., 149.,   0.],
        [109., 165.,   0.],
        ...,
        [110., 167.,   0.],
        [107., 163.,   0.],
        [134., 189.,   8.]],

       [[ 79., 125.,   0.],
        [107., 161.,  24.],
        [115., 170.,  14.],
        ...,
        [117., 178.,   0.],
        [122., 182.,   0.],
        [148., 202.,   2.]]], dtype=float32)
```

```
x.shape
```

```
(128, 128, 3)
```

```
x = np.expand_dims(x,axis=0)
x.shape
```

```
(1, 128, 128, 3)
```

```
pred_prob=model.predict(x)
pred_prob
```

```
1/1 [==============================] - 5s 5s/step
```

ASSIGNMENT 3

```
array([[0.17055237, 0.2510295 , 0.18373464, 0.17030248, 0.22438103]],
      dtype=float32)
```

```
class_name=["daisy","dandelion","rose","sunflower","tulip"]
pred_id=pred_prob.argmax(axis=1)[0]
pred_id
```

```
1
```

```
print("Predicted Flower is ",str(class_name[pred_id]))
```

```
Predicted Flower is dandelion
```