## **ASSIGNMENT-III**

# **BUILD CNN MODEL for CLASSIFICATION of FLOWERS**

Assignment Date 30 September 2022

Student Name Christal Soumia DV

Student Roll Number 963619104006

Maximum Marks 2 Marks

### Question-1:

Download the dataset

### Question-2:

**Image Augmentation** 

### Solution

from tensorflow.keras.preprocessing.image import ImageDataGenerator train\_datagen=ImageDataGenerator(rescale=1./255,zoom\_range=0.2,horizontal\_flip=True,vertical\_flip=True)

test\_datagen=ImageDataGenerator(rescale=1./255)



## Question-3:

Create model

### Solution

from tensorflow.keras.models import Sequential from tensorflow.keras.layers import Dense,Convolution2D,MaxPooling2D,Flatten model=Sequential()

3)Create Model	
[ ] from tensorflow.keras.models import Sequential	
[ ] from tensorflow.keras.layers import Dense,Convolution2D,MaxPooling2D,Flatten	
[ ] model=Sequential()	
	↑ ↓ ⊕ <b>目 /</b> ∏ î :

### Question-4:

Add Layers (Convolution, MaxPooling, Flatten, Dense-(Hidden Layers), Output)

#### Solution

# a)Convolution Layer

model.add(Convolution2D(32,(3,3),kernel\_initializer="random\_uniform",activation="relu",strides=(1,1),input\_shape=(64,64,3)))

## b) MaxPooling Layer

model.add(MaxPooling2D(pool\_size=(2,2)))

## c) Flatten Layer

model.add(Flatten())

# d) Dense(Hidden layer)

model.add(Dense(300,activation="relu")) model.add(Dense(300,activation="relu"))

## e) Output layer

model.add(Dense(5,activation="softmax"))

### Question-5:

Compile The Model

### Solution

model.compile(loss="categorical\_crossentropy",metrics=['accuracy'],optimizer='adam')

```
5)Compile the model

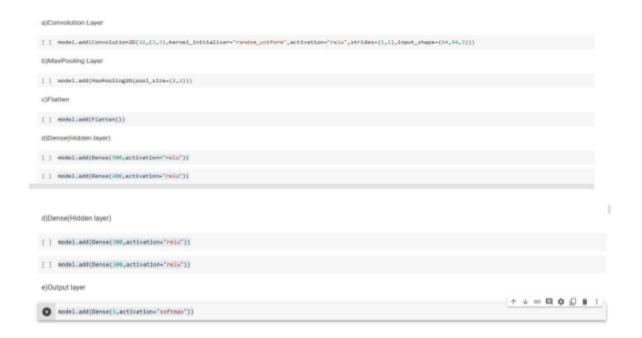
[ ] model.compile(loss="categorical_crossentropy",metrics=['accuracy'],optimizer='adam')
```

### Question-6:

Fit The Model

### Solution

model.fit(x\_train,epochs=5,steps\_per\_epoch=len(x\_train),validation\_data=x\_test,validation\_steps=len(x\_test))



### Question-7:

Save The Model

#### **Solution**

model.save("Flowers.h5")

7)Save the model
[ ] model.save("Flowers.h5")

#### Question-8:

Test The Model

# **Solution** import

numpy as np

from tensorflow.keras.models import load\_model from tensorflow.keras.preprocessing import image model=load\_model("Flowers.h5") img=image.load\_img(r"/content/drive/MyDrive/Assignment 3/Flowers-Dataset/Testing/daisy/14333681205\_a07c9f1752\_m.jpg",target\_size=(64,64)) x=image.img\_to\_array(img) x=np.expand\_dims(x,axis=0) pred=model.predict(x) pred index=['daisy','dandelion','rose','sunflower','tulip'] index[np.argmax(pred)]

	est the model
-,	
[]	<pre>import numpy as np from tensorflow.keras.models import load_model from tensorflow.keras.preprocessing import image</pre>
[]	model=load_model("Flowers.hS")
[]	img=image.load_img(r"/content/drive/MyDrive/Assignment 3/flowers-Dataset/Testing/daisy/14333681205_a07c9f1752_m.jpg",target_size=(64,64))
[]	img
[]	x=image.img_to_array(img)
r 1	x=np.expand_dims(x,axis=0)
[]	v-image ing to annaw(img)
[]	x=image.img_to_array(img)
	<pre>x=image.img_to_array(img) x=np.expand_dims(x,axis=0)</pre>
[]	
[]	x=np.expand_dims(x,axis=0)
[]	<pre>x=np.expand_dims(x, axis=0) pred=model.predict(x)</pre>
[ ] [ ]	<pre>x-np.expand_dims(x,axis=0) pred=model.predict(x) pred array([[1., 0., 0., 0., 0., 0.]], dtype=float32) index=['daisy', 'dandelion', 'rose', 'sunflower', 'tulip']</pre>
[]	<pre>x=np.expand_dims(x,axis=0) pred=model.predict(x) pred array([[1., 0., 0., 0., 0., 0.]], dtype=float32)</pre>