Assignment -3

Build CNN Model for Classification Of Flowers

| Assignment Date | 06 October 2022 |
|---------------------|-----------------|
| Student Name | Khavya P |
| Student Roll Number | 113219041051 |
| Maximum Marks | 2 Marks |

1.Download the Dataset: Dataset

Dataset downloaded and uploaded

2.Image Augmentation

```
from tensorflow.keras.preprocessing.image import ImageDataGenerator
daisy_datagen=ImageDataGenerator(rescale=1./255,zoom_range=0.5,horizont
al_flip=True,vertical_flip=True)
```

```
x_data= daisy_datagen.flow_from_directory(r"/content/drive/MyDrive/flow
ers/daisy",target_size=(64,64),class_mode="categorical",batch_size=24)
```

```
x data.class indices
```

Image Augmentation

```
[5] from tensorflow.keras.preprocessing.image import ImageDataGenerator

[12] daisy_datagen=ImageDataGenerator(rescale=1./255,zoom_range=0.5,horizontal_flip=True,vertical_flip=True)

[16] x_data= daisy_datagen.flow_from_directory(r"/content/drive/MyDrive/flowers",target_size=(64,64),class_mode="categorical",batch_size=24)

Found 52 images belonging to 5 classes.

[17] x_data.class_indices

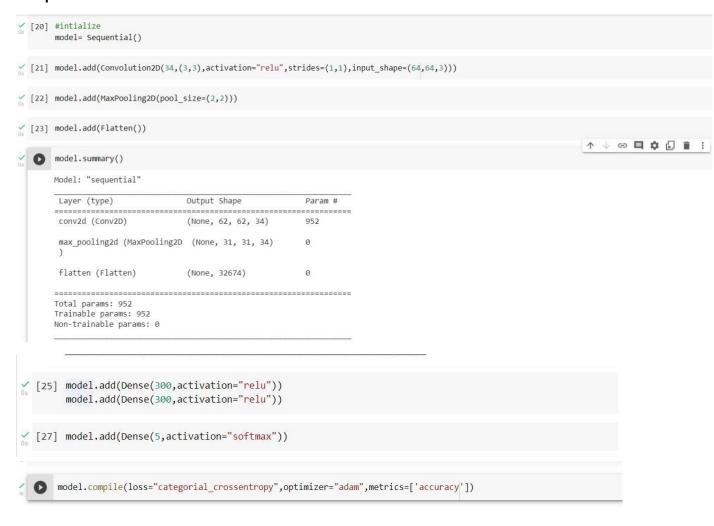
{'daisy': 0, 'dandelion': 1, 'rose': 2, 'sunflower': 3, 'tulip': 4}
```

3. Create Model

```
[18] from tensorflow.keras.models import Sequential from tensorflow.keras.layers import Dense,Convolution2D,MaxPooling2D,Flatten
```

4.Add Layers (Convolution, MaxPooling, Flatten, Dense-(Hidden Layers), Output) 5.

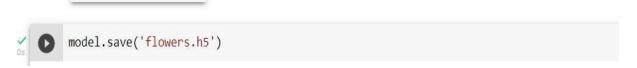
Compile The Model



6. Fit The Model

model.fit(x_data, epochs= 10, steps_per_epoch= len(x_data) ,validation_ data=0.0,validation steps=0.0)

7. Save The Model



8. Test The Model

```
[2] import numpy as np
         from tensorflow.keras.models import load_model
         from tensorflow.keras.preprocessing import image
[3] img=image.load_img(r"/content/drive/MyDrive/flower/100080576_f52e8ee070_n.jpg")
[4] img
[5] img-image.load_img(r"/content/drive/MyDrive/flower/100080576_f52e8ee070_n.jpg",target_size=(64,64))

√ [6] img

[7] x=image.img_to_array(img)
✓ D x
          array([[[141., 141., 139.], [149., 149., 149.],
                    [152., 152., 154.],
                    [162., 161., 166.],
[154., 154., 152.],
[153., 153., 153.]],
                   [[136., 135., 131.],
[146., 145., 143.],
[169., 168., 174.],
                    [159., 158., 163.],
                    [155., 155., 153.],
[149., 149., 149.]],
                   [[125., 125., 117.],
                    [138., 140., 137.],
[152.. 152.. 152.].
```

```
0
                [156., 156., 156.],
                [157., 157., 155.],
                [143., 142., 140.]],
               [[ 41., 44., 23.],
 [ 43., 46., 25.],
 [ 49., 51., 37.],
                [128., 124., 121.],
                [125., 121., 118.],
[125., 122., 117.]],
               [[ 43., 46., 25.],
                [ 43., 46., 25.],
[ 54., 55., 37.],
                [130., 126., 125.],
                [129., 125., 124.],
[127., 123., 122.]],
               [[ 44., 47., 26.],
                [ 45., 48., 27.],
[ 53., 55., 34.],
                [137., 133., 132.],
                [133., 129., 128.],
                [130., 126., 125.]]], dtype=float32)
```

x=np.expand_dims(x,axis=0) pred=
model.predict(x)

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

[23] index=['daisy', 'dandelion', 'rose', 'sunflower', 'tulip']

[24] index=[np.argmax(pred)]

index
```

 $x_{data.class_indices}$