Assignment 3

```
● Download the Dataset : https://drive.google.com/file/d
/1xkynpL15pt6KT3YSlDimu4A5iRU9qYck/view
● Image Augmentation
● Create Model
● Add Layers (Convolution,MaxPooling,Flatten,Dense-(Hidden
Layers),Output)
● Compile The Model
● Fit The Model
● Save The Model
● Test The Model
```

Image Augmentation

In [1]:

```python
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd
from tensorflow.keras.models import Sequential, load_model
from tensorflow.keras.layers import Dense,Activation,Dropout,Conv2D,Flatten,MaxP
from tensorflow.keras.applications.resnet50 import ResNet50
from tensorflow.keras.applications.resnet50 import preprocess_input
from tensorflow.keras.preprocessing import image
from tensorflow.keras.preprocessing.image import ImageDataGenerator,load_img,img
from tensorflow.keras.callbacks import EarlyStopping, ReduceLROnPlateau

import warnings
warnings.filterwarnings("ignore")
```

In [2]:

```python
train_data_gen = ImageDataGenerator(rescale = 1./255, shear_range = 0.2, zoom_ra
```

In [3]:

```python
test_data_gen = ImageDataGenerator(rescale = 1./255,validation_split = 0.30)
```

Load Data

In [4]:

```python
x_train = train_data_gen.flow_from_directory("C:/Users/santh/IBM/Flowers-Dataset
                                              target_size=(64,64),
                                              batch_size=100,
                                              class_mode='categorical',
                                              shuffle=True,
                                              color_mode='rgb',
                                              subset = 'training')
```

```
Found 3024 images belonging to 5 classes.
```

In [5]:

```python
x_test= test_data_gen.flow_from_directory("C:/Users/santh/IBM/Flowers-Dataset/fl
                                              target_size=(64,64),
                                              batch_size=100,
                                              class_mode='categorical',
                                              shuffle=True,
                                              color_mode='rgb',
                                              subset = 'validation')
```

```
Found 1293 images belonging to 5 classes.
```

```
In [6]: x_train.class_indices
```

```
Out[6]: {'daisy': 0, 'dandelion': 1, 'rose': 2, 'sunflower': 3, 'tulip': 4}
```

```
In [7]: x_test.class_indices
```

```
Out[7]: {'daisy': 0, 'dandelion': 1, 'rose': 2, 'sunflower': 3, 'tulip': 4}
```

Create Model

```
In [8]: from tensorflow.keras.models import Sequential
```

```
In [9]: from tensorflow.keras.layers import Dense,Convolution2D,MaxPooling2D,Flatten
```

```
In [10]: model=Sequential()
```

Add Layers

a)Convolution Layer

```
In [11]: model.add(Convolution2D(32,(3,3),kernel_initializer="random_uniform",activation=
```

b)MaxPooling Layer

```
In [12]: model.add(MaxPooling2D(pool_size=(2,2)))
```

c)Flatten

```
In [13]: model.add(Flatten())
```

d)Dense(Hidden layer)

```
In [14]: model.add(Dense(300,activation="relu"))
```

```
In [15]: model.add(Dense(300,activation="relu"))
```

e)Output layer

```
In [16]: model.add(Dense(5,activation="softmax"))
```

Compile the model

```
In [17]: model.compile(loss="categorical_crossentropy",metrics=['accuracy'],optimizer='ad
```

Fit the model

```
In [18]: model.fit(x_train,epochs=5,steps_per_epoch=len(x_train),validation_data=x_test,v
Epoch 1/5
31/31 [==============================] - 13s 404ms/step - loss: 1.4061 - accurac
y: 0.3935 - val_loss: 1.2636 - val_accuracy: 0.4896
Epoch 2/5
31/31 [==============================] - 11s 365ms/step - loss: 1.1463 - accurac
y: 0.5251 - val_loss: 1.2706 - val_accuracy: 0.4888
Epoch 3/5
31/31 [==============================] - 11s 343ms/step - loss: 1.0648 - accurac
y: 0.5724 - val_loss: 1.2056 - val_accuracy: 0.5019
Epoch 4/5
31/31 [==============================] - 11s 340ms/step - loss: 1.0100 - accurac
y: 0.5949 - val_loss: 1.1476 - val_accuracy: 0.5684
Epoch 5/5
31/31 [==============================] - 11s 343ms/step - loss: 0.9508 - accurac
y: 0.6333 - val_loss: 1.1851 - val_accuracy: 0.5545
```

Out[18]:

```
<keras.callbacks.History at 0x27ebb8abb50>
```

Save the model

In [19]:

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

Test the model

In [20]:

```
model=load_model("Flowers.h5")
```

In [21]:

```
img=image.load_img(r"C:/Users/santh/IBM/Flowers-Dataset/flowers/daisy/1433368120
```

In [22]:

```
img
```

Out[22]:

In [23]:

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

In [24]:

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

In [25]:

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

```
1/1 [==============================] - 0s 64ms/step
```

In [26]:

```
pred
```

Out[26]:

```
array([[0., 0., 0., 1., 0.]], dtype=float32)
```

In [27]:

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

In [28]:

```
index[np.argmax(pred)]
```

Out[28]:

```
'sunflower'
```