

Assignment -3
Python Programming

Assignment Date	12 October 2022
Project Id	PNT2022MID37914
Project Name	Natural Disaster Intensity Analysis and Classification using Artificial intelligence
Maximum Marks	2 Marks

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

```
train_datagen = ImageDataGenerator(rescale=1./255,  
                                   zoom_range=0.2,  
                                   horizontal_flip=True)
```

```
test_datagen = ImageDataGenerator(rescale=1./255)
```

```
xtrain = train_datagen.flow_from_directory('/content/drive/MyDrive/flowers',  
                                           target_size=(64,64),  
                                           class_mode='categorical',  
                                           batch_size=100)
```

Found 4317 images belonging to 5 classes.

IMAGE AUGMENTATION

From tensorflow.keras.models import Sequential

From tensorflow.keras.layers import Convolution2D,MaxPooling2D,Flatten,Dense

CNN MODEL

```
Model = Sequential()
```

```
Model.add(Convolution2D(32,(3,3),activation='relu',input_shape=(64,64,3))) # Convolution layer
```

```
Model.add(MaxPooling2D(pool_size=(2,2))) # Max pooling layer
```

```
Model.add(Flatten()) # Flatten layer
```

```
# Fully connected layers (ANN)
```

```
Model.add(Dense(300,activation='relu')) # Hidden layer 1
```

```
Model.add(Dense(150,activation='relu')) # Hidden layer 2
```

```
Model.add(Dense(4,activation='softmax')) # Output layer
```

COMPILE THE MODEL

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

SAVE THE MODEL

```
Model.save('Flower.h5')
```

TEST THE MODEL

Import numpy as np

From tensorflow.keras.preprocessing import image

```
Img=image.load_img('/content/drive/MyDrive/flowers/dandelion/10043234166_e6dd915111_n.jpg',target_size=(64,64))
```

Img

Output:



Converting image to array

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

X

Output:

```
Array([[ 3.,  6.,  0.],  
       [ 8.,  9.,  1.],  
       [ 7.,  8.,  0.],  
       ...,  
       [35., 44.,  1.]
```

[35., 44., 1.],
[35., 43., 2.]],

[[5., 8., 1.],
[6., 7., 0.],
[6., 7., 1.],
...,
[30., 36., 0.],
[30., 38., 0.],
[31., 39., 0.]],

[[7., 8., 2.],
[7., 8., 2.],
[5., 8., 0.],
...,
[19., 22., 1.],
[19., 24., 1.],
[23., 29., 3.]],

...,

[[17., 20., 1.],
[23., 27., 2.],
[24., 30., 2.],
...,
[23., 27., 0.],
[23., 29., 1.],
[21., 27., 1.]],

[[16., 19., 0.],

```

[23., 28., 0.],
[26., 31., 1.],
...,
[19., 23., 0.],
[25., 27., 5.],
[19., 24., 1.]],

[[17., 20., 1.],
[22., 26., 0.],
[26., 31., 1.],
...,
[18., 21., 0.],
[20., 24., 1.],
[21., 25., 2.]]], dtype=float32)

```

Expanding dimensions

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

```
X
```

Output:

```

Array([[[[ 3., 6., 0.],
          [ 8., 9., 1.],
          [ 7., 8., 0.],
          ...,
          [35., 44., 1.],
          [35., 44., 1.],
          [35., 43., 2.]],

```

[[5., 8., 1.],
[6., 7., 0.],
[6., 7., 1.],
...,
[30., 36., 0.],
[30., 38., 0.],
[31., 39., 0.]],

[[7., 8., 2.],
[7., 8., 2.],
[5., 8., 0.],
...,
[19., 22., 1.],
[19., 24., 1.],
[23., 29., 3.]],

...,

[[17., 20., 1.],
[23., 27., 2.],
[24., 30., 2.],
...,
[23., 27., 0.],
[23., 29., 1.],
[21., 27., 1.]],

[[16., 19., 0.],
[23., 28., 0.],
[26., 31., 1.],

```
...,
[19., 23., 0.],
[25., 27., 5.],
[19., 24., 1.]],

[[17., 20., 1.],
[22., 26., 0.],
[26., 31., 1.],
...,
[18., 21., 0.],
[20., 24., 1.],
[21., 25., 2.]]], dtype=float32)
```

```
Model.predict(x)
```

Output:

```
Array([[8.1779763e-02, 9.1822016e-01, 2.1105427e-24, 1.6366634e-27]],
      dtype=float32)
```

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

```
Pred = np.argmax(model.predict(x))
```

```
Op[pred]
```

Output:

```
'dandelion'
```

```
#Testing                               Img                               =
image.load_img('/content/drive/MyDrive/flowers/daisy/1031799732_e7f4008c03.jpg',target_size=(
64,64))

X = image.img_to_array(img)
X = np.expand_dims(x,axis=0)
Pred = np.argmax(model.predict(x))
Op[pred]
```

Output:

'dandelion'

