

```
from google.colab import drive
drive.mount('/content/drive')
```

Drive already mounted at /content/drive; to attempt to forcibly remount, call drive.m



```
ls
```

```
!unzip Animal_Dataset.zip
```

```
n#image agumentation -image data gen
```

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

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

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

+ Code

+ Text

```
x_train = train_datagen.flow_from_directory(r'dataset/Training', target_size=(64,64)
```

```
x_test = train_datagen.flow_from_directory(r'dataset/Testing', target_size=(64,64), c
```

```
x_train.class_indices
```

```
from tensorflow.keras.models import Sequential
```

```
from tensorflow.keras.layers import Dense
```

```
from tensorflow.keras.layers import Convolution2D, MaxPooling2D, Flatten
```

```
model=Sequential()
```

```
model.add(Convolution2D(32, (3,3), input_shape=(64,64,3), activation='relu'))
```

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

```
model.add(Flatten())
```

```
model.summary()
```

```
32*(3*3*3+1)
```

```

model.add(Dense(300,activation='relu'))
model.add(Dense(150,activation='relu'))

model.add(Dense(4,activation='softmax'))

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

len(x_train)

model.fit_generator(x_train,steps_per_epoch=len(x_train),validation_data=x_test,validation.

model.save('animals.h5')

```

```
ls
```

```
#testing
```

```

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

```

```
model=load_model('animals.h5')
```

```
img=image.load_img(r'dataset/Training/elephants/images (45).jpeg')
```

```
img
```



```
img=image.load_img(r'dataset/Training/elephants/images (45).jpeg',target_size=(64,64))
```

```
img
```



```
x=image.img_to_array(img)

x

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

x

y=np.argmax(model.predict(x),axis=1)
y

x_train.class_indices

index=['bears','crows','elephants','rats']

index[y[0]]

    'elephants'

#image processing
#classification,localization,detection,segmentation
```