

```
import tensorflow as tf
from tensorflow import keras
from tensorflow.keras import layers
from tensorflow.keras.preprocessing.image import ImageDataGenerator
IMG_SIZE = 244
BATCH_SIZE = 32
```

```
train_datagen = ImageDataGenerator(rescale=1./255,validation_split=0.2) # Correct the typo here
```

```
train_generator = train_datagen.flow_from_directory(
    '/content/drive/MyDrive/Eggs Classification',
    target_size=(IMG_SIZE, IMG_SIZE),
    batch_size= BATCH_SIZE,
    class_mode='binary',
    subset='training'
)
val_generator = train_datagen.flow_from_directory(
    '/content/drive/MyDrive/Eggs Classification',
    target_size=(IMG_SIZE,IMG_SIZE),
    batch_size=BATCH_SIZE,
    class_mode='binary',
    subset='validation'
)
```

Found 636 images belonging to 2 classes.  
Found 158 images belonging to 2 classes.

```
model = keras.Sequential([
    layers.Conv2D(32, (3,3),activation='relu',input_shape=(IMG_SIZE,IMG_SIZE,3)),
    layers.MaxPooling2D(2,2),
    layers.Conv2D(64,(3,3),activation='relu'),
    layers.MaxPooling2D(2,2),
    layers.Conv2D(128,(3,3),activation='relu'),
    layers.MaxPooling2D(2,2),
    layers.Flatten(),
    layers.Dense(128,activation='relu'),
    layers.Dense(1,activation='sigmoid') #output layer
])
```

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

```
model.fit(train_generator,validation_data=val_generator,epochs=5)
```

Epoch 1/5  
20/20 [=====] - 198s 10s/step - loss: 0.7381 - accuracy: 0.7217 - val\_loss: 0.5197 - val\_accu  
Epoch 2/5  
20/20 [=====] - 106s 5s/step - loss: 0.4653 - accuracy: 0.8066 - val\_loss: 0.5138 - val\_accu  
Epoch 3/5  
20/20 [=====] - 112s 6s/step - loss: 0.3915 - accuracy: 0.8412 - val\_loss: 0.5316 - val\_accu  
Epoch 4/5

```
model.save("Model.h5","label.txt")
```

/usr/local/lib/python3.10/dist-packages/keras/src/engine/training.py:3103: UserWarning: You are saving your model as a saving\_api.save\_model(

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

```
model = load_model('/content/Model.h5')
test_image_path = ('/content/drive/MyDrive/Eggs Classification/Not Damaged/not_damaged_1.jpg')
img = image.load_img(test_image_path, target_size=(244, 244)) # Change target size to 244x244
img_array = image.img_to_array(img)
img_array = np.expand_dims(img_array, axis=0)
```

```
img_array = img_array / 255.0
```

```
predictions = model.predict(img_array)
print(predictions)
```

```
1/1 [=====] - 0s 182ms/step
[[0.12986402]]
```

```
if predictions < 0.5:
    print('It is a Damaged')
else:
    print('It is a Not Damaged')
```

```
It is a Damaged
```

The screenshot displays a Google Colab notebook environment. The top navigation bar includes tabs for 'day1\_017.ipynb - Colab', 'MIT App Inventor', 'Home - Google Drive', and '017 - Google Docs'. The address bar shows the URL: [colab.research.google.com/drive/1sPZjlaWcqW5ApF1r58ABqrCumATiqwqe#scrollTo=uL64WfFthRj0](https://colab.research.google.com/drive/1sPZjlaWcqW5ApF1r58ABqrCumATiqwqe#scrollTo=uL64WfFthRj0). The notebook interface features a left sidebar with a file explorer showing a directory structure: 'drive' > 'MyDrive' > 'Eggs Classification' > 'Damaged'. The main area is divided into a code editor and an image viewer. The code editor contains the following Python code:

```
import numpy as np

model = load_model('/content/Model.h5')
test_image_path = ('/content/drive/MyDrive/Eggs Classification/Not Damaged/n
img = image.load_img(test_image_path, target_size=(244, 244)) # Change targ
img_array = image.img_to_array(img)
img_array = np.expand_dims(img_array, axis=0)

img_array = img_array / 255.0

predictions = model.predict(img_array)
print(predictions)
```

Below the code editor, the output shows the execution progress: '1/1 [=====] - 0s 182ms/step' and the prediction result: '[[0.12986402]]'. A second code block shows the conditional logic:

```
if predictions < 0.5:
    print('It is a Damaged')
else:
    print('It is a Not Damaged')
```

The output of this block is 'It is a Damaged'. To the right of the code editor, an image viewer displays a photograph of a cracked egg in a nest, labeled 'damaged\_1.jpg'. The bottom status bar indicates 'Connected to Python 3 Google Compute Engine backend' and shows system information like 'NIFTY +0.18%', '10:16', and '26-06-2024'.