

Double-click (or enter) to edit

Build a CNN model for Bird species

Dataset Link: <https://www.kaggle.com/datasets/akash2907/bird-species-classification>

```
! pip install kaggle
! mkdir ~/.kaggle
! cp kaggle.json ~/.kaggle/
! chmod 600 ~/.kaggle/kaggle.json
```

```
Looking in indexes: https://pypi.org/simple, https://us-python.pkg.dev/colab-wheels/public/simple/
Requirement already satisfied: kaggle in /usr/local/lib/python3.10/dist-packages (1.5.13)
Requirement already satisfied: six>=1.10 in /usr/local/lib/python3.10/dist-packages (from kaggle) (1.16.0)
Requirement already satisfied: certifi in /usr/local/lib/python3.10/dist-packages (from kaggle) (2022.12.7)
Requirement already satisfied: python-dateutil in /usr/local/lib/python3.10/dist-packages (from kaggle) (2.8.2)
Requirement already satisfied: requests in /usr/local/lib/python3.10/dist-packages (from kaggle) (2.27.1)
Requirement already satisfied: tqdm in /usr/local/lib/python3.10/dist-packages (from kaggle) (4.65.0)
Requirement already satisfied: python-slugify in /usr/local/lib/python3.10/dist-packages (from kaggle) (8.0.1)
Requirement already satisfied: urllib3 in /usr/local/lib/python3.10/dist-packages (from kaggle) (1.26.15)
Requirement already satisfied: text-unidecode>=1.3 in /usr/local/lib/python3.10/dist-packages (from python-slugify->kaggle) (1.3)
Requirement already satisfied: charset-normalizer~>=2.0.0 in /usr/local/lib/python3.10/dist-packages (from requests->kaggle) (2.0.12)
Requirement already satisfied: idna<4,>=2.5 in /usr/local/lib/python3.10/dist-packages (from requests->kaggle) (3.4)
mkdir: cannot create directory '/root/.kaggle': File exists
```

```
! kaggle datasets download akash2907/bird-species-classification
```

```
Downloading bird-species-classification.zip to /content
100% 1.37G/1.37G [00:53<00:00, 27.3MB/s]
100% 1.37G/1.37G [00:53<00:00, 27.4MB/s]
```

```
!unzip '/content/bird-species-classification.zip'
```

```
Archive: /content/bird-species-classification.zip
inflating: test_data/test_data/blast/DSC_6396.jpg
inflating: test_data/test_data/blast/DSC_6397.jpg
inflating: test_data/test_data/blast/DSC_6398.jpg
inflating: test_data/test_data/blast/DSC_6399.jpg
inflating: test_data/test_data/blast/DSC_6400.jpg
inflating: test_data/test_data/blast/DSC_6401.jpg
inflating: test_data/test_data/blast/DSC_6402.jpg
inflating: test_data/test_data/blast/DSC_6403.jpg
inflating: test_data/test_data/blast/DSC_6405.jpg
inflating: test_data/test_data/blast/DSC_6406.jpg
inflating: test_data/test_data/blast/DSC_6407.jpg
inflating: test_data/test_data/blast/DSC_6408.jpg
inflating: test_data/test_data/blast/DSC_6409.jpg
inflating: test_data/test_data/blast/DSC_6410.jpg
inflating: test_data/test_data/blast/DSC_6411.jpg
inflating: test_data/test_data/bonegl/DSC_4587.jpg
inflating: test_data/test_data/bonegl/DSC_4588.jpg
inflating: test_data/test_data/bonegl/DSC_4589.jpg
inflating: test_data/test_data/bonegl/DSC_4590.jpg
inflating: test_data/test_data/bonegl/DSC_4591.jpg
inflating: test_data/test_data/bonegl/DSC_4592.jpg
inflating: test_data/test_data/bonegl/DSC_4593.jpg
inflating: test_data/test_data/brhkyt/D72_0473.jpg
inflating: test_data/test_data/brhkyt/D72_0474.jpg
inflating: test_data/test_data/brhkyt/D72_0475.jpg
inflating: test_data/test_data/brhkyt/D72_0477.jpg
inflating: test_data/test_data/brhkyt/D72_0478.jpg
inflating: test_data/test_data/brhkyt/D72_0479.jpg
inflating: test_data/test_data/cbrtsh/_D32_10310.jpg
inflating: test_data/test_data/cbrtsh/_D32_10311.jpg
inflating: test_data/test_data/cbrtsh/_D32_10312.jpg
inflating: test_data/test_data/cbrtsh/_D32_10313.jpg
inflating: test_data/test_data/cbrtsh/_D32_10314.jpg
inflating: test_data/test_data/cbrtsh/_D32_10317.jpg
inflating: test_data/test_data/cbrtsh/_D32_10318.jpg
inflating: test_data/test_data/cnmyn/DSC_2443.jpg
inflating: test_data/test_data/cnmyn/DSC_4681.jpg
inflating: test_data/test_data/cnmyn/DSC_5137.jpg
inflating: test_data/test_data/cnmyn/DSC_7625.jpg
inflating: test_data/test_data/cnmyn/P1050277.jpg
inflating: test_data/test_data/cnmyn/_D32_12426.jpg
inflating: test_data/test_data/cnmyn/_D32_12427.jpg
inflating: test_data/test_data/cnmyn/_D32_12428.jpg
inflating: test_data/test_data/gretit/11620454726_31a35c26da_o.jpg
inflating: test_data/test_data/gretit/11776135285_ccf938fa2e_o.jpg
inflating: test_data/test_data/gretit/11905645146_6a5d4ff9f9_o.jpg
inflating: test_data/test_data/gretit/8537646712_0b282c4c6a_o.jpg
inflating: test_data/test_data/gretit/D72_0693.jpg
inflating: test_data/test_data/gretit/D72_0694.jpg
```

```
inflating: test_data/test_data/gretit/D72_0695.jpg
inflating: test_data/test_data/hilpig/DSC_6359.jpg
inflating: test_data/test_data/hilpig/DSC_6362.jpg
inflating: test_data/test_data/hilpig/DSC_6364.jpg
inflating: test_data/test_data/hilpig/DSC_6368.jpg
inflating: test_data/test_data/hilpig/DSC_6403.jpg
inflating: test_data/test_data/hilpig/DSC_6404.jpg
inflating: test_data/test_data/hilpig/P1000319.jpg
```

```
import os
import random
import numpy as np
import tensorflow as tf
from tensorflow import keras
from tensorflow.keras.preprocessing.image import ImageDataGenerator
```

```
# Define the paths to the train and test datasets
train_data_dir = '/content/train_data'
test_data_dir = '/content/test_data'
```

```
# Data preprocessing and augmentation for training dataset
train_datagen = ImageDataGenerator(rescale=(1./255), horizontal_flip=True, shear_range=0.5)
```

```
train_generator = train_datagen.flow_from_directory(
    train_data_dir,
    target_size=(224,224),
    batch_size=4,
    class_mode='categorical'
)
```

Found 150 images belonging to 1 classes.

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

```
test_generator = test_datagen.flow_from_directory(
    test_data_dir,
    target_size=(224, 224),
    batch_size=4,
    class_mode='categorical',
    shuffle=False)
```

Found 157 images belonging to 1 classes.

```
model = keras.Sequential([
    keras.layers.Conv2D(32, (3, 3), activation='relu', input_shape=(img_width, img_height, 3)),
    keras.layers.MaxPooling2D((2, 2)),
    keras.layers.Conv2D(64, (3, 3), activation='relu'),
    keras.layers.MaxPooling2D((2, 2)),
    keras.layers.Conv2D(128, (3, 3), activation='relu'),
    keras.layers.MaxPooling2D((2, 2)),
    keras.layers.Flatten(),
    keras.layers.Dense(128, activation='relu'),
    keras.layers.Dropout(0.5),
    keras.layers.Dense(num_classes, activation='softmax')
])
```

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

```
# Train the model
model.fit(
    train_generator,
    steps_per_epoch=train_generator.samples // 4,
    epochs=10
```

```
)

# Evaluate the model on the test dataset
test_loss, test_acc = model.evaluate(test_generator)
print(f'Test Loss: {test_loss}')
print(f'Test Accuracy: {test_acc}')
```

Epoch 1/10
/usr/local/lib/python3.10/dist-packages/tensorflow/python/util/dispatch.py:1176: SyntaxWarning: In loss categorical_crossentropy, €
return dispatch_target(*args, **kwargs)
37/37 [=====] - 71s 2s/step - loss: 0.0000e+00 - accuracy: 1.0000
Epoch 2/10
37/37 [=====] - 64s 2s/step - loss: 0.0000e+00 - accuracy: 1.0000
Epoch 3/10
37/37 [=====] - 64s 2s/step - loss: 0.0000e+00 - accuracy: 1.0000
Epoch 4/10
37/37 [=====] - 64s 2s/step - loss: 0.0000e+00 - accuracy: 1.0000
Epoch 5/10
37/37 [=====] - 62s 2s/step - loss: 0.0000e+00 - accuracy: 1.0000
Epoch 6/10
37/37 [=====] - 62s 2s/step - loss: 0.0000e+00 - accuracy: 1.0000
Epoch 7/10
37/37 [=====] - 64s 2s/step - loss: 0.0000e+00 - accuracy: 1.0000
Epoch 8/10
37/37 [=====] - 63s 2s/step - loss: 0.0000e+00 - accuracy: 1.0000
Epoch 9/10
37/37 [=====] - 69s 2s/step - loss: 0.0000e+00 - accuracy: 1.0000
Epoch 10/10
37/37 [=====] - 64s 2s/step - loss: 0.0000e+00 - accuracy: 1.0000
40/40 [=====] - 65s 2s/step - loss: 0.0000e+00 - accuracy: 1.0000
Test Loss: 0.0
Test Accuracy: 1.0

[Colab paid products](#) - [Cancel contracts here](#)

Could not connect to the reCAPTCHA service. Please check your internet connection and reload to get a reCAPTCHA challenge.