

```
In [1]: import numpy as np
import matplotlib.pyplot as plt

import tensorflow as tf
from keras.applications.vgg19 import VGG19
from keras.layers import Flatten, Dense, Dropout
from keras.layers import GlobalAveragePooling2D
from keras.optimizers import Adam
from keras.models import Model
from keras.preprocessing.image import ImageDataGenerator
```

Using TensorFlow backend.

```
In [2]: # create an image generator
data_gen = ImageDataGenerator(rescale = 1./255, validation_split = 0.2)
```

```
In [4]: from keras.preprocessing.image import ImageDataGenerator

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

test_datagen = ImageDataGenerator(rescale = 1./255)

training_set = train_datagen.flow_from_directory("C:/Users/Project/Desktop/flowers/
train_data",
                                                target_size = (224, 224),
                                                batch_size = 32,
                                                class_mode = 'categorical')

test_set = test_datagen.flow_from_directory("C:/Users/Project/Desktop/flowers/test_
data",
                                            target_size = (224, 224),
                                            batch_size = 32,
                                            class_mode = 'categorical')
```

Found 3823 images belonging to 5 classes.

Found 500 images belonging to 5 classes.

```
In [ ]: # create the base pre-trained model
base_model = VGG19(weights="imagenet", input_shape = (224, 224, 3))

for layer in base_model.layers:
    layer.trainable = False

X = base_model.output
X = GlobalAveragePooling2D()(X)
X = Dense(128, activation = "relu")(X)
X = Dropout(0.5)(X)
X = Dense(32, activation = "relu")(X)
predictions = Dense(5, activation = "softmax")(X)

model = Model(inputs = base_model.input, outputs = predictions)
```

Downloading data from https://github.com/fchollet/deep-learning-models/releases/download/v0.1/vgg19_weights_tf_dim_ordering_tf_kernels.h5
554852352/574710816 [=====>..] - ETA: 23s

```
In [ ]: model.compile(loss = "categorical_crossentropy",
                    optimizer = "adam",
                    metrics = ["accuracy"]
                    )
```

```
In [8]: train_history = model.fit_generator(training_set, steps_per_epoch = 10, epochs = 10
)
```

WARNING:tensorflow:From D:\ProgramData\Anaconda3\lib\site-packages\tensorflow\python\ops\math_ops.py:3066: to_int32 (from tensorflow.python.ops.math_ops) is deprecated and will be removed in a future version.

Instructions for updating:

Use tf.cast instead.

Epoch 1/10

10/10 [=====] - 124s 12s/step - loss: 1.6828 - acc: 0.2875

Epoch 2/10

10/10 [=====] - 122s 12s/step - loss: 1.5616 - acc: 0.2938

Epoch 3/10

10/10 [=====] - 123s 12s/step - loss: 1.5069 - acc: 0.3594

Epoch 4/10

10/10 [=====] - 122s 12s/step - loss: 1.4813 - acc: 0.4062

Epoch 5/10

10/10 [=====] - 127s 13s/step - loss: 1.4634 - acc: 0.4156

Epoch 6/10

10/10 [=====] - 123s 12s/step - loss: 1.3936 - acc: 0.4813

Epoch 7/10

10/10 [=====] - 127s 13s/step - loss: 1.3266 - acc: 0.4844

Epoch 8/10

10/10 [=====] - 124s 12s/step - loss: 1.2943 - acc: 0.5062

Epoch 9/10

10/10 [=====] - 132s 13s/step - loss: 1.2303 - acc: 0.5500

Epoch 10/10

10/10 [=====] - 133s 13s/step - loss: 1.2060 - acc: 0.5031

In []: