In [1]:
```
import tensorflow as tf
from tensorflow.keras import models, layers
import matplotlib.pyplot as plt
```

In [2]:
```
IMAGE_SIZE = 224
BATCH_SIZE = 32
CHANNELS = 3
```

In [3]:
```
dataset =  tf.keras.preprocessing.image_dataset_from_directory(
    "PlantVillage",
    shuffle=True,
    batch_size = BATCH_SIZE,
    image_size = (IMAGE_SIZE, IMAGE_SIZE)

)
```

```
Found 16011 files belonging to 10 classes.
```

In [4]:
```
classes = dataset.class_names
```

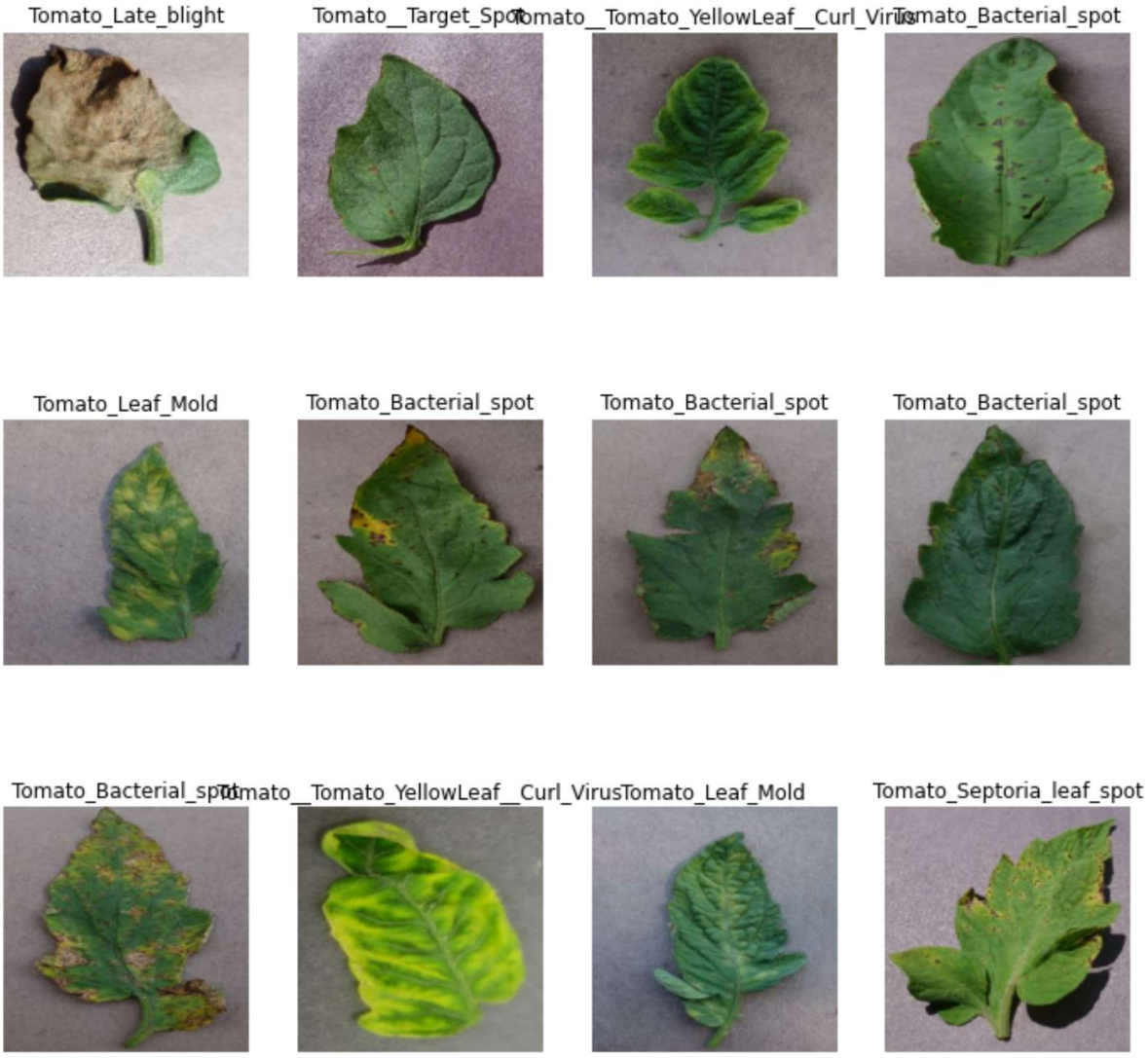
In [5]:
```
len(dataset)
```

Out[5]:
```
501
```

In [6]:
```
dataset
```

Out[6]:
```
<BatchDataset element_spec=(TensorSpec(shape=(None, 224, 224, 3), dtype=tf.float32, name=None), TensorSpec(shape=(None,), dtype=tf.int32, name=None))>
```

In [7]:
```
plt.figure(figsize=(12,12))
for image_batch, label_batch in dataset.take(1):
    for i in range(12) :
        plt.subplot(3,4,i+1)
        plt.imshow(image_batch[i].numpy().astype("uint8"))
        plt.title(classes[label_batch[i]])
        plt.axis("off")
```



In [8]:
```
def train_val_test(dataset, train_split = 0.8, val_split = 0.1, test_split = 0.1, sh

    if shuffle:
        dataset.shuffle(shuffle_size, seed = 12)

    dataset_len = len(dataset)
    train_size = int(dataset_len*train_split)
    val_size = int(dataset_len*val_split)
    test_size = int(dataset_len*test_split)

    train_ds =  dataset.take(train_size)
    val_ds = dataset.skip(train_size).take(val_size)
    test_ds = dataset.skip(train_size).skip(val_size)

    return train_ds, val_ds, test_ds
```

In [9]:
```
train_ds, val_ds, test_ds = train_val_test(dataset)
```

In [10]:
```
len(train_ds)
```

Out[10]:
```
400
```

In [11]:
```
train_ds = train_ds.cache().shuffle(1000).prefetch(tf.data.AUTOTUNE)
val_ds = val_ds.cache().shuffle(1000).prefetch(tf.data.AUTOTUNE)
test_ds = test_ds.cache().shuffle(1000).prefetch(tf.data.AUTOTUNE)
```

In [12]:
```
resize_rescale = tf.keras.Sequential([
```

```
    layers.experimental.preprocessing.Resizing(IMAGE_SIZE,IMAGE_SIZE),
    layers.experimental.preprocessing.Rescaling(1.0/255)
])
```

In [13]:
```
data_aug = tf.keras.Sequential(
    [
        tf.keras.layers.experimental.preprocessing.RandomFlip("horizontal_and_vertic
        tf.keras.layers.experimental.preprocessing.RandomRotation(0.1),
        tf.keras.layers.experimental.preprocessing.RandomZoom(0.1),
        tf.keras.layers.experimental.preprocessing.RandomContrast(0.2),
        tf.keras.layers.experimental.preprocessing.RandomHeight(0.2),


    ]
)
```

In [24]:
```
from keras.applications.vgg19 import VGG19
from keras.models import Model
from keras.layers import Input, Lambda, Dense, Flatten
```

In [21]:
```
vgg = VGG19(input_shape=[IMAGE_SIZE,IMAGE_SIZE] + [3], weights='imagenet', include_t

for layer in vgg.layers:
    layer.trainable = False
```

In [28]:
```
input_shape = (BATCH_SIZE, IMAGE_SIZE, IMAGE_SIZE, CHANNELS)
n_classes = 3

model = models.Sequential([
    resize_rescale,

    vgg,
    layers.Flatten(),
    layers.Dense(64, activation='relu'),
    layers.Dense(n_classes, activation='softmax'),
])

model.build(input_shape=input_shape)
```

our layers - you can add more if you want

x = Flatten()(vgg.output)

dense_1 = Dense(1000, activation='relu')(x) dense_2 = Dense(64, activation='relu')(dense_1) prediction = Dense(n_classes, activation = 'softmax')

create a model object

model = Model(inputs=vgg.input, outputs=prediction)

In [17]:
```
model.summary()
```

```
Model: "sequential_2"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 sequential (Sequential)     (32, 224, 224, 3)         0

 sequential_1 (Sequential)   (None, None, 224, 3)      0

 vgg19 (Functional)          (None, 7, 7, 512)         20024384

 flatten (Flatten)           (None, 25088)             0

 dense (Dense)               (None, 64)                1605696

 dense_1 (Dense)             (None, 3)                 195

=================================================================
Total params: 21,630,275
Trainable params: 1,605,891
Non-trainable params: 20,024,384
_________________________________________________________________
```

In [29]:
```
model.compile(
    optimizer='adam',
    loss=tf.keras.losses.SparseCategoricalCrossentropy(from_logits=False),
    metrics=['accuracy']
)
```

In [30]:
```
history = model.fit(
    train_ds,
    batch_size=BATCH_SIZE,
    validation_data=val_ds,
    verbose=1,
    epochs=50,
)
```

```
Epoch 1/50
```

```
---------------------------------------------------------------------------
InvalidArgumentError                      Traceback (most recent call last)
Input In [30], in <cell line: 1>()
----> 1 history = model.fit(
      2     train_ds,
      3     batch_size=BATCH_SIZE,
      4     validation_data=val_ds,
      5     verbose=1,
      6     epochs=50,
      7 )

File ~\AppData\Local\Programs\Python\Python310\lib\site-packages\keras\utils\traceba
ck_utils.py:67, in filter_traceback.<locals>.error_handler(*args, **kwargs)
     65 except Exception as e:  # pylint: disable=broad-except
     66   filtered_tb = _process_traceback_frames(e.__traceback__)
---> 67   raise e.with_traceback(filtered_tb) from None
     68 finally:
     69   del filtered_tb

File ~\AppData\Local\Programs\Python\Python310\lib\site-packages\tensorflow\python\e
ager\execute.py:54, in quick_execute(op_name, num_outputs, inputs, attrs, ctx, name)
     52 try:
     53   ctx.ensure_initialized()
---> 54   tensors = pywrap_tfe.TFE_Py_Execute(ctx._handle, device_name, op_name,
     55                                       inputs, attrs, num_outputs)
     56 except core._NotOkStatusException as e:
     57   if name is not None:

InvalidArgumentError: Graph execution error:

Detected at node 'sparse_categorical_crossentropy/SparseSoftmaxCrossEntropyWithLogit
s/SparseSoftmaxCrossEntropyWithLogits' defined at (most recent call last):
    File "C:\Users\udain\AppData\Local\Programs\Python\Python310\lib\runpy.py", line
196, in _run_module_as_main
      return _run_code(code, main_globals, None,
    File "C:\Users\udain\AppData\Local\Programs\Python\Python310\lib\runpy.py", line
86, in _run_code
      exec(code, run_globals)
    File "C:\Users\udain\AppData\Local\Programs\Python\Python310\lib\site-packages\i
pykernel_launcher.py", line 17, in <module>
      app.launch_new_instance()
    File "C:\Users\udain\AppData\Local\Programs\Python\Python310\lib\site-packages\t
raitlets\config\application.py", line 972, in launch_instance
      app.start()
    File "C:\Users\udain\AppData\Local\Programs\Python\Python310\lib\site-packages\i
pykernel\kernelapp.py", line 712, in start
      self.io_loop.start()
    File "C:\Users\udain\AppData\Local\Programs\Python\Python310\lib\site-packages\t
ornado\platform\asyncio.py", line 199, in start
      self.asyncio_loop.run_forever()
    File "C:\Users\udain\AppData\Local\Programs\Python\Python310\lib\asyncio\base_ev
ents.py", line 600, in run_forever
      self._run_once()
    File "C:\Users\udain\AppData\Local\Programs\Python\Python310\lib\asyncio\base_ev
ents.py", line 1896, in _run_once
      handle._run()
    File "C:\Users\udain\AppData\Local\Programs\Python\Python310\lib\asyncio\events.
py", line 80, in _run
      self._context.run(self._callback, *self._args)
    File "C:\Users\udain\AppData\Local\Programs\Python\Python310\lib\site-packages\i
pykernel\kernelbase.py", line 504, in dispatch_queue
      await self.process_one()
    File "C:\Users\udain\AppData\Local\Programs\Python\Python310\lib\site-packages\i
pykernel\kernelbase.py", line 493, in process_one
```

```
      await dispatch(*args)
    File "C:\Users\udain\AppData\Local\Programs\Python\Python310\lib\site-packages\i
pykernel\kernelbase.py", line 400, in dispatch_shell
      await result
    File "C:\Users\udain\AppData\Local\Programs\Python\Python310\lib\site-packages\i
pykernel\kernelbase.py", line 724, in execute_request
      reply_content = await reply_content
    File "C:\Users\udain\AppData\Local\Programs\Python\Python310\lib\site-packages\i
pykernel\ipkernel.py", line 383, in do_execute
      res = shell.run_cell(
    File "C:\Users\udain\AppData\Local\Programs\Python\Python310\lib\site-packages\i
pykernel\zmqshell.py", line 528, in run_cell
      return super().run_cell(*args, **kwargs)
    File "C:\Users\udain\AppData\Local\Programs\Python\Python310\lib\site-packages\I
Python\core\interactiveshell.py", line 2880, in run_cell
      result = self._run_cell(
    File "C:\Users\udain\AppData\Local\Programs\Python\Python310\lib\site-packages\I
Python\core\interactiveshell.py", line 2935, in _run_cell
      return runner(coro)
    File "C:\Users\udain\AppData\Local\Programs\Python\Python310\lib\site-packages\I
Python\core\async_helpers.py", line 129, in _pseudo_sync_runner
      coro.send(None)
    File "C:\Users\udain\AppData\Local\Programs\Python\Python310\lib\site-packages\I
Python\core\interactiveshell.py", line 3134, in run_cell_async
      has_raised = await self.run_ast_nodes(code_ast.body, cell_name,
    File "C:\Users\udain\AppData\Local\Programs\Python\Python310\lib\site-packages\I
Python\core\interactiveshell.py", line 3337, in run_ast_nodes
      if await self.run_code(code, result, async_=asy):
    File "C:\Users\udain\AppData\Local\Programs\Python\Python310\lib\site-packages\I
Python\core\interactiveshell.py", line 3397, in run_code
      exec(code_obj, self.user_global_ns, self.user_ns)
    File "C:\Users\udain\AppData\Local\Temp\ipykernel_1736\1320925133.py", line 1, i
n <cell line: 1>
      history = model.fit(
    File "C:\Users\udain\AppData\Local\Programs\Python\Python310\lib\site-packages\k
eras\utils\traceback_utils.py", line 64, in error_handler
      return fn(*args, **kwargs)
    File "C:\Users\udain\AppData\Local\Programs\Python\Python310\lib\site-packages\k
eras\engine\training.py", line 1409, in fit
      tmp_logs = self.train_function(iterator)
    File "C:\Users\udain\AppData\Local\Programs\Python\Python310\lib\site-packages\k
eras\engine\training.py", line 1051, in train_function
      return step_function(self, iterator)
    File "C:\Users\udain\AppData\Local\Programs\Python\Python310\lib\site-packages\k
eras\engine\training.py", line 1040, in step_function
      outputs = model.distribute_strategy.run(run_step, args=(data,))
    File "C:\Users\udain\AppData\Local\Programs\Python\Python310\lib\site-packages\k
eras\engine\training.py", line 1030, in run_step
      outputs = model.train_step(data)
    File "C:\Users\udain\AppData\Local\Programs\Python\Python310\lib\site-packages\k
eras\engine\training.py", line 890, in train_step
      loss = self.compute_loss(x, y, y_pred, sample_weight)
    File "C:\Users\udain\AppData\Local\Programs\Python\Python310\lib\site-packages\k
eras\engine\training.py", line 948, in compute_loss
      return self.compiled_loss(
    File "C:\Users\udain\AppData\Local\Programs\Python\Python310\lib\site-packages\k
eras\engine\compile_utils.py", line 201, in __call__
      loss_value = loss_obj(y_t, y_p, sample_weight=sw)
    File "C:\Users\udain\AppData\Local\Programs\Python\Python310\lib\site-packages\k
eras\losses.py", line 139, in __call__
      losses = call_fn(y_true, y_pred)
    File "C:\Users\udain\AppData\Local\Programs\Python\Python310\lib\site-packages\k
eras\losses.py", line 243, in call
      return ag_fn(y_true, y_pred, **self._fn_kwargs)
```

```
    File "C:\Users\udain\AppData\Local\Programs\Python\Python310\lib\site-packages\keras\losses.py", line 1860, in sparse_categorical_crossentropy
      return backend.sparse_categorical_crossentropy(
    File "C:\Users\udain\AppData\Local\Programs\Python\Python310\lib\site-packages\keras\backend.py", line 5238, in sparse_categorical_crossentropy
      res = tf.nn.sparse_softmax_cross_entropy_with_logits(
Node: 'sparse_categorical_crossentropy/SparseSoftmaxCrossEntropyWithLogits/SparseSoftmaxCrossEntropyWithLogits'
Received a label value of 9 which is outside the valid range of [0, 3).  Label values: 3 7 0 7 3 9 2 0 4 9 0 3 7 4 6 8 9 4 7 0 5 7 0 2 9 9 5 9 6 0 2 0
       [[{{node sparse_categorical_crossentropy/SparseSoftmaxCrossEntropyWithLogits/SparseSoftmaxCrossEntropyWithLogits}}]] [Op:__inference_train_function_10081]
```

In [ ]: