```python
import numpy as np # linear algebra
import pandas as pd # data processing, CSV file I/O (e.g. pd.read_csv)
import os

#for dirname, _, filenames in os.walk('/kaggle/input'):
#    for filename in filenames:
#        print(os.path.join(dirname, filename))

%matplotlib inline
import matplotlib.pyplot as plt
from PIL import Image
import tensorflow as tf
print(tf.__version__)
```

```
2.3.1
```

Ignoring the wildlife.h5 file as not needed for this exercise.

Investigate the dataset

- How many of each class of animal do we have

In [2]:

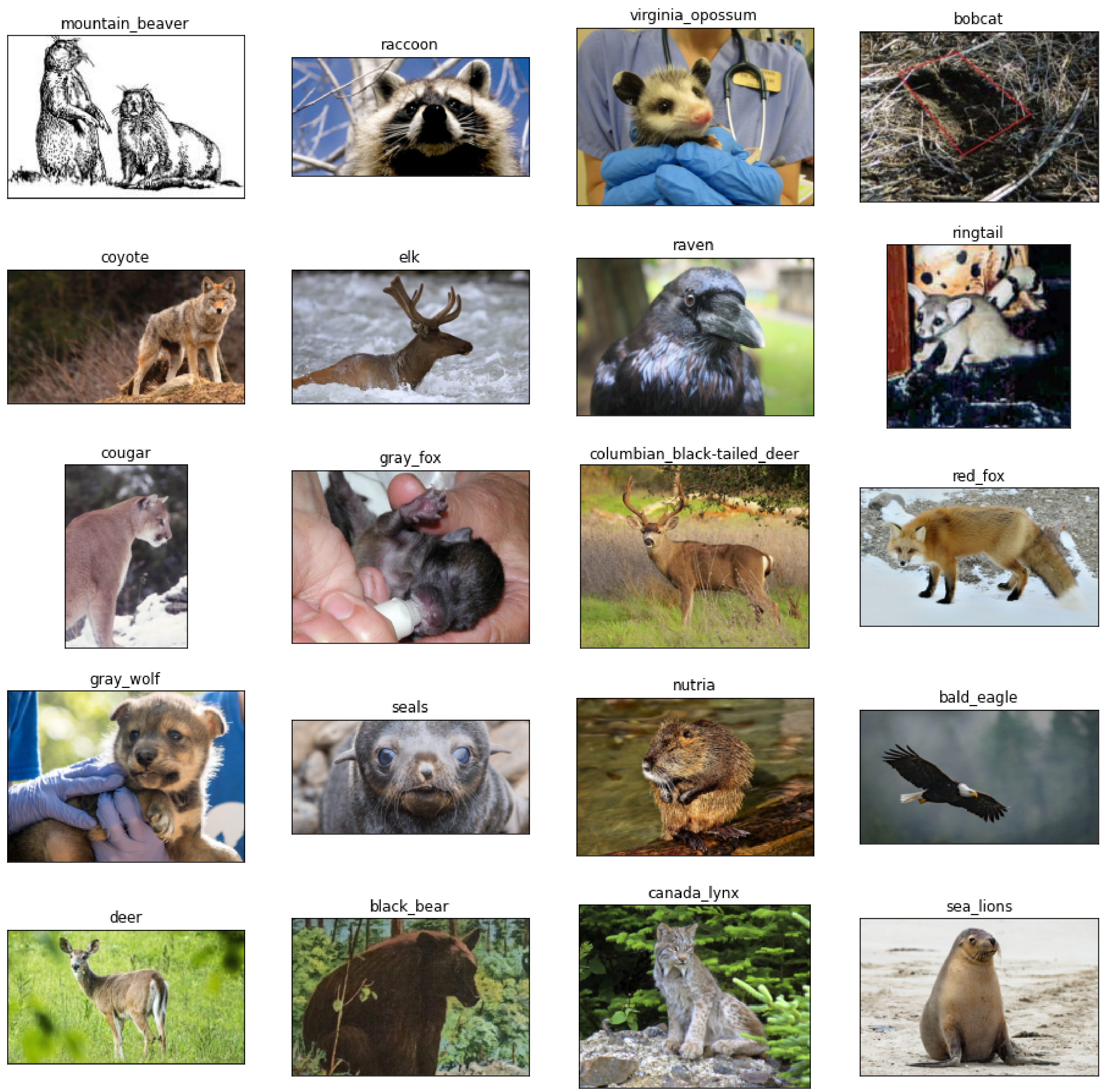
```
PATH = '../input/oregon-wildlife/oregon_wildlife/oregon_wildlife/'
animal_list = os.listdir(PATH)

for animal in animal_list:
    number = len(os.listdir(PATH+animal))
    print('There are ', number, animal, 'images')
print('There are ', len(animal_list), 'total categories')
```

```
There are  577 mountain_beaver images
There are  728 raccoon images
There are  728 virginia_opossum images
There are  696 bobcat images
There are  736 coyote images
There are  660 elk images
There are  656 raven images
There are  588 ringtail images
There are  680 cougar images
There are  668 gray_fox images
There are  735 columbian_black-tailed_deer images
There are  759 red_fox images
There are  730 gray_wolf images
There are  698 seals images
There are  701 nutria images
There are  748 bald_eagle images
There are  764 deer images
There are  718 black_bear images
There are  717 canada_lynx images
There are  726 sea_lions images
There are  20 total categories
```

Now visualize an image from each category ¶

In [3]:

```python
fig=plt.figure(figsize=(16, 16))
columns = 4
rows = 5
i=0
for animal in animal_list:
    i+=1
    file = os.listdir(PATH+animal)[0]
    img = Image.open(PATH+animal+'/'+file)
    fig.add_subplot(rows, columns, i, xticks=[], yticks=[])
    plt.title(animal)
    plt.imshow(img)
```



Load and split the data in to test, train, validation sets

Use Keras.preprocessing.image_dataset_from_directory to generate train, test and validation sets. The training and validation steps are from: https://github.com/tensorflow/docs/blob/master/site/en/tutorials/load_data/images.ipynb

In [4]:

```
#Set the batch size, and image height and width
batch_size = 32
img_height = 224
img_width = 224
IMG_SIZE = (img_height, img_width)
```

In [5]:

```
#Generate the training dataset
train_ds = tf.keras.preprocessing.image_dataset_from_directory(
  PATH,
  validation_split=0.2,
  subset="training",
  label_mode='int',
  seed=123,
  image_size=(img_height, img_width),
  batch_size=batch_size)
```

```
Found 14013 files belonging to 20 classes.
Using 11211 files for training.
```

In [6]:

```
#Generate the validation dataset
val_ds = tf.keras.preprocessing.image_dataset_from_directory(
  PATH,
  validation_split=0.2,
  subset="validation",
  label_mode='int',
  seed=123,
  image_size=(img_height, img_width),
  batch_size=batch_size)
```

```
Found 14013 files belonging to 20 classes.
Using 2802 files for validation.
```

Since there was not test set create one from the validation dataset. In this case we are finding the cardinality `tf.data.experimental.cardinality` of the validation dataset (val_ds). Then create a test dataset from that by using the take `tf.data.Dataset.take()` method and 20% (val_batches // 5) where the '//' is a floor division. Then takes the unused elements of val_ds to remain in the val_ds.

In [7]:
```
val_batches = tf.data.experimental.cardinality(val_ds)
test_ds = val_ds.take(val_batches // 5)
val_sd = val_ds.skip(val_batches // 5)
```

Verification that the class names are valid and there are the correct number.

In [8]:
```
class_names = train_ds.class_names
num_classes = len(class_names)
print('Class names are: ', class_names, '/n The number of classes is: ', num_classes)
```

```
Class names are:  ['bald_eagle', 'black_bear', 'bobcat', 'canada_lynx', 'columbian_blac
k-tailed_deer', 'cougar', 'coyote', 'deer', 'elk', 'gray_fox', 'gray_wolf', 'mountain_b
eaver', 'nutria', 'raccoon', 'raven', 'red_fox', 'ringtail', 'sea_lions', 'seals', 'vir
ginia_opossum'] /n The number of classes is:  20
```

1. In the next few steps we will load the `tf.keras.applications.MobileNetV2` base model. This model needs to be preprocessed with values between [-1,1] so we will process the inputs to it with `tf.keras.applications.mobilenet_v2.preprocess_input` .

In [9]:
```
from tensorflow.keras import layers

preprocess_input = tf.keras.applications.mobilenet_v2.preprocess_input
```

Setup buffering and prefetching of the training and validation datasets

In [10]:

```
AUTOTUNE = tf.data.experimental.AUTOTUNE  #This sets up the runtime to dynamically tune the
buffersize

train_ds = train_ds.cache().prefetch(buffer_size=AUTOTUNE)
val_ds = val_ds.cache().prefetch(buffer_size=AUTOTUNE)
```

- **Here I'm loading the MobileNetV2 model from Keras. I'm not including the top layer, so I can create a cutom classification layer.**

In [11]:

```
# Create the base model from the pre-trained model MobileNet V2
IMG_SHAPE = IMG_SIZE + (3,)
base_model = tf.keras.applications.MobileNetV2(input_shape=IMG_SHAPE,
                                               include_top=False,
                                               weights='imagenet')
```

```
Downloading data from https://storage.googleapis.com/tensorflow/keras-applications/mobi
lenet_v2/mobilenet_v2_weights_tf_dim_ordering_tf_kernels_1.0_224_no_top.h5
9412608/9406464 [==============================] - 0s 0us/step
```

This feature extractor converts each 160x160x3 image into a n x n x p block of features in order to always be able to get the correct size for the new feature extration layer we'll measure the shape of a batch of data from the train_dataset. The output will be a tensor of shape (batch_size, n, n, p).

In [12]:

```
image_batch, label_batch = next(iter(train_ds))
feature_batch = base_model(image_batch)
print(feature_batch.shape)
```

```
(32, 7, 7, 1280)
```

Feature Extraction

In this section we freeze the base model layers so the pre-learned weights and biases aren't updated during training and then add the new classifier on top and train.

In [13]:

```
base_model.trainable = False
```

The base model layers need to be frozen, this way the weights in the layers, except for the top layer won't be updated during training. The top layer is not included in order to add a custom classifier layer with the 20 classes we're training on. See this TensorFlow Tutorial. As we'll see MobilNet contains batch normalization layers, and special care should be taken as described here: BatchNorm Description

In [14]:

```
base_model.summary()  #Look at the model architecture
```

```
Model: "mobilenetv2_1.00_224"
__________________________________________________________________________________________________
Layer (type)                    Output Shape         Param #     Connected to
==================================================================================================
input_1 (InputLayer)            [(None, 224, 224, 3) 0
__________________________________________________________________________________________________
Conv1_pad (ZeroPadding2D)       (None, 225, 225, 3)  0           input_1[0][0]
__________________________________________________________________________________________________
Conv1 (Conv2D)                  (None, 112, 112, 32) 864         Conv1_pad[0][0]
__________________________________________________________________________________________________
bn_Conv1 (BatchNormalization)   (None, 112, 112, 32) 128         Conv1[0][0]
__________________________________________________________________________________________________
Conv1_relu (ReLU)               (None, 112, 112, 32) 0           bn_Conv1[0][0]
__________________________________________________________________________________________________
expanded_conv_depthwise (Depthw (None, 112, 112, 32) 288         Conv1_relu[0][0]
__________________________________________________________________________________________________
expanded_conv_depthwise_BN (Bat (None, 112, 112, 32) 128         expanded_conv_depthwis
e[0][0]
__________________________________________________________________________________________________
expanded_conv_depthwise_relu (R (None, 112, 112, 32) 0           expanded_conv_depthwis
e_BN[0][0]
__________________________________________________________________________________________________
expanded_conv_project (Conv2D)  (None, 112, 112, 16) 512         expanded_conv_depthwis
e_relu[0][0
__________________________________________________________________________________________________
expanded_conv_project_BN (Batch (None, 112, 112, 16) 64          expanded_conv_project
[0][0]
__________________________________________________________________________________________________
block_1_expand (Conv2D)         (None, 112, 112, 96) 1536        expanded_conv_project_
BN[0][0]
__________________________________________________________________________________________________
block_1_expand_BN (BatchNormali (None, 112, 112, 96) 384         block_1_expand[0][0]
```

```
______________________________________________________________________________________
____________
block_1_expand_relu (ReLU)      (None, 112, 112, 96) 0           block_1_expand_BN[0]
[0]
______________________________________________________________________________________
____________
block_1_pad (ZeroPadding2D)     (None, 113, 113, 96) 0           block_1_expand_relu[0]
[0]
______________________________________________________________________________________
____________
block_1_depthwise (DepthwiseCon (None, 56, 56, 96)   864         block_1_pad[0][0]
______________________________________________________________________________________
____________
block_1_depthwise_BN (BatchNorm (None, 56, 56, 96)   384         block_1_depthwise[0]
[0]
______________________________________________________________________________________
____________
block_1_depthwise_relu (ReLU)   (None, 56, 56, 96)   0           block_1_depthwise_BN
[0][0]
______________________________________________________________________________________
____________
block_1_project (Conv2D)        (None, 56, 56, 24)   2304        block_1_depthwise_relu
[0][0]
______________________________________________________________________________________
____________
block_1_project_BN (BatchNormal (None, 56, 56, 24)   96          block_1_project[0][0]
______________________________________________________________________________________
____________
block_2_expand (Conv2D)         (None, 56, 56, 144)  3456        block_1_project_BN[0]
[0]
______________________________________________________________________________________
____________
block_2_expand_BN (BatchNormali (None, 56, 56, 144)  576         block_2_expand[0][0]
______________________________________________________________________________________
____________
block_2_expand_relu (ReLU)      (None, 56, 56, 144)  0           block_2_expand_BN[0]
[0]
______________________________________________________________________________________
____________
block_2_depthwise (DepthwiseCon (None, 56, 56, 144)  1296        block_2_expand_relu[0]
[0]
______________________________________________________________________________________
____________
block_2_depthwise_BN (BatchNorm (None, 56, 56, 144)  576         block_2_depthwise[0]
[0]
______________________________________________________________________________________
```

```
___________
block_2_depthwise_relu (ReLU)   (None, 56, 56, 144)  0           block_2_depthwise_BN
[0][0]
__________________________________________________________________________________________________
___________
block_2_project (Conv2D)        (None, 56, 56, 24)   3456        block_2_depthwise_relu
[0][0]
__________________________________________________________________________________________________
___________
block_2_project_BN (BatchNormal (None, 56, 56, 24)   96          block_2_project[0][0]
__________________________________________________________________________________________________
___________
block_2_add (Add)               (None, 56, 56, 24)   0           block_1_project_BN[0]
[0]
                                                                 block_2_project_BN[0]
[0]
__________________________________________________________________________________________________
___________
block_3_expand (Conv2D)         (None, 56, 56, 144)  3456        block_2_add[0][0]
__________________________________________________________________________________________________
___________
block_3_expand_BN (BatchNormali (None, 56, 56, 144)  576         block_3_expand[0][0]
__________________________________________________________________________________________________
___________
block_3_expand_relu (ReLU)      (None, 56, 56, 144)  0           block_3_expand_BN[0]
[0]
__________________________________________________________________________________________________
___________
block_3_pad (ZeroPadding2D)     (None, 57, 57, 144)  0           block_3_expand_relu[0]
[0]
__________________________________________________________________________________________________
___________
block_3_depthwise (DepthwiseCon (None, 28, 28, 144)  1296        block_3_pad[0][0]
__________________________________________________________________________________________________
___________
block_3_depthwise_BN (BatchNorm (None, 28, 28, 144)  576         block_3_depthwise[0]
[0]
__________________________________________________________________________________________________
___________
block_3_depthwise_relu (ReLU)   (None, 28, 28, 144)  0           block_3_depthwise_BN
[0][0]
__________________________________________________________________________________________________
___________
block_3_project (Conv2D)        (None, 28, 28, 32)   4608        block_3_depthwise_relu
[0][0]
__________________________________________________________________________________________________
```

```
___________
block_3_project_BN (BatchNormal (None, 28, 28, 32)   128         block_3_project[0][0]
___________
block_4_expand (Conv2D)         (None, 28, 28, 192)  6144        block_3_project_BN[0]
[0]
___________
block_4_expand_BN (BatchNormali (None, 28, 28, 192)  768         block_4_expand[0][0]
___________
block_4_expand_relu (ReLU)      (None, 28, 28, 192)  0           block_4_expand_BN[0]
[0]
___________
block_4_depthwise (DepthwiseCon (None, 28, 28, 192)  1728        block_4_expand_relu[0]
[0]
___________
block_4_depthwise_BN (BatchNorm (None, 28, 28, 192)  768         block_4_depthwise[0]
[0]
___________
block_4_depthwise_relu (ReLU)   (None, 28, 28, 192)  0           block_4_depthwise_BN
[0][0]
___________
block_4_project (Conv2D)        (None, 28, 28, 32)   6144        block_4_depthwise_relu
[0][0]
___________
block_4_project_BN (BatchNormal (None, 28, 28, 32)   128         block_4_project[0][0]
___________
block_4_add (Add)               (None, 28, 28, 32)   0           block_3_project_BN[0]
[0]
                                                                 block_4_project_BN[0]
[0]
___________
block_5_expand (Conv2D)         (None, 28, 28, 192)  6144        block_4_add[0][0]
___________
block_5_expand_BN (BatchNormali (None, 28, 28, 192)  768         block_5_expand[0][0]
___________
```

```
block_5_expand_relu (ReLU)      (None, 28, 28, 192)  0           block_5_expand_BN[0]
[0]
_______________________________________________________________________________________
__________
block_5_depthwise (DepthwiseCon (None, 28, 28, 192)  1728        block_5_expand_relu[0]
[0]
_______________________________________________________________________________________
__________
block_5_depthwise_BN (BatchNorm (None, 28, 28, 192)  768         block_5_depthwise[0]
[0]
_______________________________________________________________________________________
__________
block_5_depthwise_relu (ReLU)   (None, 28, 28, 192)  0           block_5_depthwise_BN
[0][0]
_______________________________________________________________________________________
__________
block_5_project (Conv2D)        (None, 28, 28, 32)   6144        block_5_depthwise_relu
[0][0]
_______________________________________________________________________________________
__________
block_5_project_BN (BatchNormal (None, 28, 28, 32)   128         block_5_project[0][0]
_______________________________________________________________________________________
__________
block_5_add (Add)               (None, 28, 28, 32)   0           block_4_add[0][0]
                                                                 block_5_project_BN[0]
[0]
_______________________________________________________________________________________
__________
block_6_expand (Conv2D)         (None, 28, 28, 192)  6144        block_5_add[0][0]
_______________________________________________________________________________________
__________
block_6_expand_BN (BatchNormali (None, 28, 28, 192)  768         block_6_expand[0][0]
_______________________________________________________________________________________
__________
block_6_expand_relu (ReLU)      (None, 28, 28, 192)  0           block_6_expand_BN[0]
[0]
_______________________________________________________________________________________
__________
block_6_pad (ZeroPadding2D)     (None, 29, 29, 192)  0           block_6_expand_relu[0]
[0]
_______________________________________________________________________________________
__________
block_6_depthwise (DepthwiseCon (None, 14, 14, 192)  1728        block_6_pad[0][0]
_______________________________________________________________________________________
__________
block_6_depthwise_BN (BatchNorm (None, 14, 14, 192)  768         block_6_depthwise[0]
```

```
[0]
__________________________________________________________________________________________________
block_6_depthwise_relu (ReLU)   (None, 14, 14, 192)  0           block_6_depthwise_BN
[0][0]
__________________________________________________________________________________________________
block_6_project (Conv2D)        (None, 14, 14, 64)   12288       block_6_depthwise_relu
[0][0]
__________________________________________________________________________________________________
block_6_project_BN (BatchNormal (None, 14, 14, 64)   256         block_6_project[0][0]
__________________________________________________________________________________________________
block_7_expand (Conv2D)         (None, 14, 14, 384)  24576       block_6_project_BN[0]
[0]
__________________________________________________________________________________________________
block_7_expand_BN (BatchNormali (None, 14, 14, 384)  1536        block_7_expand[0][0]
__________________________________________________________________________________________________
block_7_expand_relu (ReLU)      (None, 14, 14, 384)  0           block_7_expand_BN[0]
[0]
__________________________________________________________________________________________________
block_7_depthwise (DepthwiseCon (None, 14, 14, 384)  3456        block_7_expand_relu[0]
[0]
__________________________________________________________________________________________________
block_7_depthwise_BN (BatchNorm (None, 14, 14, 384)  1536        block_7_depthwise[0]
[0]
__________________________________________________________________________________________________
block_7_depthwise_relu (ReLU)   (None, 14, 14, 384)  0           block_7_depthwise_BN
[0][0]
__________________________________________________________________________________________________
block_7_project (Conv2D)        (None, 14, 14, 64)   24576       block_7_depthwise_relu
[0][0]
__________________________________________________________________________________________________
block_7_project_BN (BatchNormal (None, 14, 14, 64)   256         block_7_project[0][0]
__________________________________________________________________________________________________
block_7_add (Add)               (None, 14, 14, 64)   0           block_6_project_BN[0]
[0]
```

```
                                                                  block_7_project_BN[0]
[0]
 ________________________________________________________________________________________
 ___________
 block_8_expand (Conv2D)         (None, 14, 14, 384)  24576       block_7_add[0][0]
 ________________________________________________________________________________________
 ___________
 block_8_expand_BN (BatchNormali (None, 14, 14, 384)  1536        block_8_expand[0][0]
 ________________________________________________________________________________________
 ___________
 block_8_expand_relu (ReLU)      (None, 14, 14, 384)  0           block_8_expand_BN[0]
[0]
 ________________________________________________________________________________________
 ___________
 block_8_depthwise (DepthwiseCon (None, 14, 14, 384)  3456        block_8_expand_relu[0]
[0]
 ________________________________________________________________________________________
 ___________
 block_8_depthwise_BN (BatchNorm (None, 14, 14, 384)  1536        block_8_depthwise[0]
[0]
 ________________________________________________________________________________________
 ___________
 block_8_depthwise_relu (ReLU)   (None, 14, 14, 384)  0           block_8_depthwise_BN
[0][0]
 ________________________________________________________________________________________
 ___________
 block_8_project (Conv2D)        (None, 14, 14, 64)   24576       block_8_depthwise_relu
[0][0]
 ________________________________________________________________________________________
 ___________
 block_8_project_BN (BatchNormal (None, 14, 14, 64)   256         block_8_project[0][0]
 ________________________________________________________________________________________
 ___________
 block_8_add (Add)               (None, 14, 14, 64)   0           block_7_add[0][0]
                                                                  block_8_project_BN[0]
[0]
 ________________________________________________________________________________________
 ___________
 block_9_expand (Conv2D)         (None, 14, 14, 384)  24576       block_8_add[0][0]
 ________________________________________________________________________________________
 ___________
 block_9_expand_BN (BatchNormali (None, 14, 14, 384)  1536        block_9_expand[0][0]
 ________________________________________________________________________________________
 ___________
 block_9_expand_relu (ReLU)      (None, 14, 14, 384)  0           block_9_expand_BN[0]
[0]
```

```
__________________________________________________________________________________________________
block_9_depthwise (DepthwiseCon (None, 14, 14, 384)  3456        block_9_expand_relu[0]
[0]
__________________________________________________________________________________________________
block_9_depthwise_BN (BatchNorm (None, 14, 14, 384)  1536        block_9_depthwise[0]
[0]
__________________________________________________________________________________________________
block_9_depthwise_relu (ReLU)   (None, 14, 14, 384)  0           block_9_depthwise_BN
[0][0]
__________________________________________________________________________________________________
block_9_project (Conv2D)        (None, 14, 14, 64)   24576       block_9_depthwise_relu
[0][0]
__________________________________________________________________________________________________
block_9_project_BN (BatchNormal (None, 14, 14, 64)   256         block_9_project[0][0]
__________________________________________________________________________________________________
block_9_add (Add)               (None, 14, 14, 64)   0           block_8_add[0][0]
                                                                 block_9_project_BN[0]
[0]
__________________________________________________________________________________________________
block_10_expand (Conv2D)        (None, 14, 14, 384)  24576       block_9_add[0][0]
__________________________________________________________________________________________________
block_10_expand_BN (BatchNormal (None, 14, 14, 384)  1536        block_10_expand[0][0]
__________________________________________________________________________________________________
block_10_expand_relu (ReLU)     (None, 14, 14, 384)  0           block_10_expand_BN[0]
[0]
__________________________________________________________________________________________________
block_10_depthwise (DepthwiseCo (None, 14, 14, 384)  3456        block_10_expand_relu
[0][0]
__________________________________________________________________________________________________
block_10_depthwise_BN (BatchNor (None, 14, 14, 384)  1536        block_10_depthwise[0]
[0]
__________________________________________________________________________________________________
block_10_depthwise_relu (ReLU)  (None, 14, 14, 384)  0           block_10_depthwise_BN
[0][0]
```

```
____________
block_10_project (Conv2D)       (None, 14, 14, 96)   36864       block_10_depthwise_rel
u[0][0]
________________________________________________________________________________________
____________
block_10_project_BN (BatchNorma (None, 14, 14, 96)   384         block_10_project[0][0]
________________________________________________________________________________________
____________
block_11_expand (Conv2D)        (None, 14, 14, 576)  55296       block_10_project_BN[0]
[0]
________________________________________________________________________________________
____________
block_11_expand_BN (BatchNormal (None, 14, 14, 576)  2304        block_11_expand[0][0]
________________________________________________________________________________________
____________
block_11_expand_relu (ReLU)     (None, 14, 14, 576)  0           block_11_expand_BN[0]
[0]
________________________________________________________________________________________
____________
block_11_depthwise (DepthwiseCo (None, 14, 14, 576)  5184        block_11_expand_relu
[0][0]
________________________________________________________________________________________
____________
block_11_depthwise_BN (BatchNor (None, 14, 14, 576)  2304        block_11_depthwise[0]
[0]
________________________________________________________________________________________
____________
block_11_depthwise_relu (ReLU)  (None, 14, 14, 576)  0           block_11_depthwise_BN
[0][0]
________________________________________________________________________________________
____________
block_11_project (Conv2D)       (None, 14, 14, 96)   55296       block_11_depthwise_rel
u[0][0]
________________________________________________________________________________________
____________
block_11_project_BN (BatchNorma (None, 14, 14, 96)   384         block_11_project[0][0]
________________________________________________________________________________________
____________
block_11_add (Add)              (None, 14, 14, 96)   0           block_10_project_BN[0]
[0]
                                                                 block_11_project_BN[0]
[0]
________________________________________________________________________________________
____________
block_12_expand (Conv2D)        (None, 14, 14, 576)  55296       block_11_add[0][0]
```

```
________________________________________________________________________________________

__________
block_12_expand_BN (BatchNormal (None, 14, 14, 576)  2304        block_12_expand[0][0]
________________________________________________________________________________________

__________
block_12_expand_relu (ReLU)     (None, 14, 14, 576)  0           block_12_expand_BN[0]
[0]
________________________________________________________________________________________

__________
block_12_depthwise (DepthwiseCo (None, 14, 14, 576)  5184        block_12_expand_relu
[0][0]
________________________________________________________________________________________

__________
block_12_depthwise_BN (BatchNor (None, 14, 14, 576)  2304        block_12_depthwise[0]
[0]
________________________________________________________________________________________

__________
block_12_depthwise_relu (ReLU)  (None, 14, 14, 576)  0           block_12_depthwise_BN
[0][0]
________________________________________________________________________________________

__________
block_12_project (Conv2D)       (None, 14, 14, 96)   55296       block_12_depthwise_rel
u[0][0]
________________________________________________________________________________________

__________
block_12_project_BN (BatchNorma (None, 14, 14, 96)   384         block_12_project[0][0]
________________________________________________________________________________________

__________
block_12_add (Add)              (None, 14, 14, 96)   0           block_11_add[0][0]
                                                                 block_12_project_BN[0]

[0]
________________________________________________________________________________________

__________
block_13_expand (Conv2D)        (None, 14, 14, 576)  55296       block_12_add[0][0]
________________________________________________________________________________________

__________
block_13_expand_BN (BatchNormal (None, 14, 14, 576)  2304        block_13_expand[0][0]
________________________________________________________________________________________

__________
block_13_expand_relu (ReLU)     (None, 14, 14, 576)  0           block_13_expand_BN[0]
[0]
________________________________________________________________________________________

__________
block_13_pad (ZeroPadding2D)    (None, 15, 15, 576)  0           block_13_expand_relu
[0][0]
________________________________________________________________________________________
```

```
__________
block_13_depthwise (DepthwiseCo (None, 7, 7, 576)    5184        block_13_pad[0][0]
________________________________________________________________________________
__________
block_13_depthwise_BN (BatchNor (None, 7, 7, 576)    2304        block_13_depthwise[0]
[0]
________________________________________________________________________________
__________
block_13_depthwise_relu (ReLU)  (None, 7, 7, 576)    0           block_13_depthwise_BN
[0][0]
________________________________________________________________________________
__________
block_13_project (Conv2D)       (None, 7, 7, 160)    92160       block_13_depthwise_rel
u[0][0]
________________________________________________________________________________
__________
block_13_project_BN (BatchNorma (None, 7, 7, 160)    640         block_13_project[0][0]
________________________________________________________________________________
__________
block_14_expand (Conv2D)        (None, 7, 7, 960)    153600      block_13_project_BN[0]
[0]
________________________________________________________________________________
__________
block_14_expand_BN (BatchNormal (None, 7, 7, 960)    3840        block_14_expand[0][0]
________________________________________________________________________________
__________
block_14_expand_relu (ReLU)     (None, 7, 7, 960)    0           block_14_expand_BN[0]
[0]
________________________________________________________________________________
__________
block_14_depthwise (DepthwiseCo (None, 7, 7, 960)    8640        block_14_expand_relu
[0][0]
________________________________________________________________________________
__________
block_14_depthwise_BN (BatchNor (None, 7, 7, 960)    3840        block_14_depthwise[0]
[0]
________________________________________________________________________________
__________
block_14_depthwise_relu (ReLU)  (None, 7, 7, 960)    0           block_14_depthwise_BN
[0][0]
________________________________________________________________________________
__________
block_14_project (Conv2D)       (None, 7, 7, 160)    153600      block_14_depthwise_rel
u[0][0]
________________________________________________________________________________
__________
```

```
block_14_project_BN (BatchNorma (None, 7, 7, 160)    640         block_14_project[0][0]
__________________________________________________________________________________________
__________
block_14_add (Add)              (None, 7, 7, 160)    0           block_13_project_BN[0]
[0]
                                                                 block_14_project_BN[0]
[0]
__________________________________________________________________________________________
__________
block_15_expand (Conv2D)        (None, 7, 7, 960)    153600      block_14_add[0][0]
__________________________________________________________________________________________
__________
block_15_expand_BN (BatchNormal (None, 7, 7, 960)    3840        block_15_expand[0][0]
__________________________________________________________________________________________
__________
block_15_expand_relu (ReLU)     (None, 7, 7, 960)    0           block_15_expand_BN[0]
[0]
__________________________________________________________________________________________
__________
block_15_depthwise (DepthwiseCo (None, 7, 7, 960)    8640        block_15_expand_relu
[0][0]
__________________________________________________________________________________________
__________
block_15_depthwise_BN (BatchNor (None, 7, 7, 960)    3840        block_15_depthwise[0]
[0]
__________________________________________________________________________________________
__________
block_15_depthwise_relu (ReLU)  (None, 7, 7, 960)    0           block_15_depthwise_BN
[0][0]
__________________________________________________________________________________________
__________
block_15_project (Conv2D)       (None, 7, 7, 160)    153600      block_15_depthwise_rel
u[0][0]
__________________________________________________________________________________________
__________
block_15_project_BN (BatchNorma (None, 7, 7, 160)    640         block_15_project[0][0]
__________________________________________________________________________________________
__________
block_15_add (Add)              (None, 7, 7, 160)    0           block_14_add[0][0]
                                                                 block_15_project_BN[0]
[0]
__________________________________________________________________________________________
__________
block_16_expand (Conv2D)        (None, 7, 7, 960)    153600      block_15_add[0][0]
__________________________________________________________________________________________
__________
```

```
block_16_expand_BN (BatchNormal (None, 7, 7, 960)    3840        block_16_expand[0][0]
__________________________________________________________________________________________________
block_16_expand_relu (ReLU)     (None, 7, 7, 960)    0           block_16_expand_BN[0][0]
__________________________________________________________________________________________________
block_16_depthwise (DepthwiseCo (None, 7, 7, 960)    8640        block_16_expand_relu[0][0]
__________________________________________________________________________________________________
block_16_depthwise_BN (BatchNor (None, 7, 7, 960)    3840        block_16_depthwise[0][0]
__________________________________________________________________________________________________
block_16_depthwise_relu (ReLU)  (None, 7, 7, 960)    0           block_16_depthwise_BN[0][0]
__________________________________________________________________________________________________
block_16_project (Conv2D)       (None, 7, 7, 320)    307200      block_16_depthwise_relu[0][0]
__________________________________________________________________________________________________
block_16_project_BN (BatchNorma (None, 7, 7, 320)    1280        block_16_project[0][0]
__________________________________________________________________________________________________
Conv_1 (Conv2D)                 (None, 7, 7, 1280)   409600      block_16_project_BN[0][0]
__________________________________________________________________________________________________
Conv_1_bn (BatchNormalization)  (None, 7, 7, 1280)   5120        Conv_1[0][0]
__________________________________________________________________________________________________
out_relu (ReLU)                 (None, 7, 7, 1280)   0           Conv_1_bn[0][0]
==================================================================================================
Total params: 2,257,984
Trainable params: 0
Non-trainable params: 2,257,984
__________________________________________________________________________________________________
```

Help and inputs at this step are from Adding a Classification Head In order to use the output to create a classification layer the output of the base model is, n,n,p. Need to convert this to a single 1280 element vector so this is done by using a average pooling layer.

In [15]:

```
global_average_layer = tf.keras.layers.GlobalAveragePooling2D()
feature_batch_average = global_average_layer(feature_batch)
prediction_layer = tf.keras.layers.Dense(num_classes, activation='softmax', name='predictio
n')
prediction_batch = prediction_layer(feature_batch_average)
print(prediction_batch.shape)
```

```
(32, 20)
```

In [16]:

```
data_augmentation = tf.keras.Sequential([
  tf.keras.layers.experimental.preprocessing.RandomFlip('horizontal'),
  tf.keras.layers.experimental.preprocessing.RandomRotation(0.2),
])
```

In [17]:

```
inputs = tf.keras.Input(shape=(img_height, img_width, 3))
x = data_augmentation(inputs)
x = preprocess_input(inputs)
x = base_model(x)
x = global_average_layer(x)
x = tf.keras.layers.Dropout(0.2)(x)
outputs = prediction_layer(x)
model = tf.keras.models.Model(inputs, outputs)
```

Compile the Model

Compile the model before training. There are two classes, uses sparse_catgegorical_crossentropy

In [18]:

```
base_learning_rate = 0.0001
model.compile(optimizer=tf.keras.optimizers.Adam(lr=base_learning_rate),
              loss=tf.keras.losses.SparseCategoricalCrossentropy(from_logits=True),
              metrics=['accuracy'])
```

In [19]:

```
model.summary()
```

```
Model: "functional_1"
_________________________________________________________________
Layer (type)                 Output Shape              Param #   
=================================================================
input_2 (InputLayer)         [(None, 224, 224, 3)]     0         
_________________________________________________________________
tf_op_layer_RealDiv (TensorF [(None, 224, 224, 3)]     0         
_________________________________________________________________
tf_op_layer_Sub (TensorFlowO [(None, 224, 224, 3)]     0         
_________________________________________________________________
mobilenetv2_1.00_224 (Functi (None, 7, 7, 1280)        2257984   
_________________________________________________________________
global_average_pooling2d (Gl (None, 1280)              0         
_________________________________________________________________
dropout (Dropout)            (None, 1280)              0         
_________________________________________________________________
prediction (Dense)           (None, 20)                25620     
=================================================================
Total params: 2,283,604
Trainable params: 25,620
Non-trainable params: 2,257,984
_________________________________________________________________
```

Train the model

Setup initial training parameters and evaluation criterial

In [20]:

```
epochs = 10
```

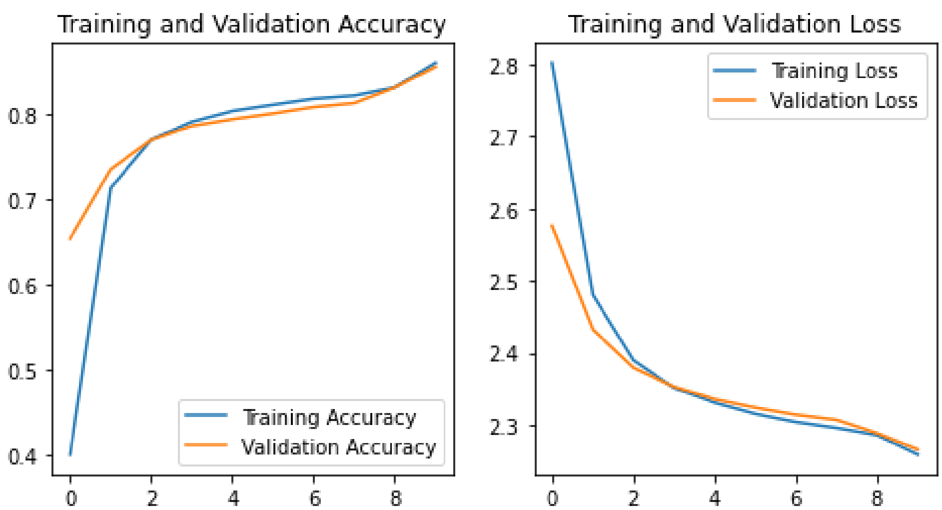
In [21]:

```
history = model.fit(train_ds,
                    validation_data=val_ds,
                    epochs=epochs)
```

```
Epoch 1/10
351/351 [==============================] - 288s 820ms/step - loss: 2.8024 - accuracy:
0.4012 - val_loss: 2.5764 - val_accuracy: 0.6542
Epoch 2/10
351/351 [==============================] - 12s 35ms/step - loss: 2.4804 - accuracy: 0.7
135 - val_loss: 2.4323 - val_accuracy: 0.7355
Epoch 3/10
351/351 [==============================] - 12s 35ms/step - loss: 2.3897 - accuracy: 0.7
700 - val_loss: 2.3794 - val_accuracy: 0.7698
Epoch 4/10
351/351 [==============================] - 13s 36ms/step - loss: 2.3513 - accuracy: 0.7
911 - val_loss: 2.3528 - val_accuracy: 0.7862
Epoch 5/10
351/351 [==============================] - 12s 35ms/step - loss: 2.3309 - accuracy: 0.8
038 - val_loss: 2.3357 - val_accuracy: 0.7941
Epoch 6/10
351/351 [==============================] - 12s 35ms/step - loss: 2.3153 - accuracy: 0.8
111 - val_loss: 2.3239 - val_accuracy: 0.8009
Epoch 7/10
351/351 [==============================] - 12s 35ms/step - loss: 2.3039 - accuracy: 0.8
180 - val_loss: 2.3141 - val_accuracy: 0.8084
Epoch 8/10
351/351 [==============================] - 12s 35ms/step - loss: 2.2958 - accuracy: 0.8
217 - val_loss: 2.3069 - val_accuracy: 0.8130
Epoch 9/10
351/351 [==============================] - 13s 36ms/step - loss: 2.2858 - accuracy: 0.8
313 - val_loss: 2.2885 - val_accuracy: 0.8312
Epoch 10/10
351/351 [==============================] - 12s 35ms/step - loss: 2.2589 - accuracy: 0.8
601 - val_loss: 2.2658 - val_accuracy: 0.8555
```

In [22]:

```
acc = history.history['accuracy']
val_acc = history.history['val_accuracy']

loss = history.history['loss']
val_loss = history.history['val_loss']

epochs_range = range(epochs)

plt.figure(figsize=(8, 4))
plt.subplot(1, 2, 1)
plt.plot(epochs_range, acc, label='Training Accuracy')
plt.plot(epochs_range, val_acc, label='Validation Accuracy')
plt.legend(loc='lower right')
plt.title('Training and Validation Accuracy')

plt.subplot(1, 2, 2)
plt.plot(epochs_range, loss, label='Training Loss')
plt.plot(epochs_range, val_loss, label='Validation Loss')
plt.legend(loc='upper right')
plt.title('Training and Validation Loss')
plt.show()
```



Baseline Model Performance

The baseline model, which contained no augmentation or fine tuning had a validation dataset accuracy of 85%.

In [23]:

```
#This will evaluate the baseline model against the test dataset.
test_loss, test_accuracy = model.evaluate(test_ds)
print('Test dataset loss: ', test_loss, 'Test dataset accuracy: ', test_accuracy)
```

```
17/17 [==============================] - 6s 349ms/step - loss: 2.2477 - accuracy: 0.873
2
Test dataset loss:  2.2477364540100098 Test dataset accuracy:  0.873161792755127
```

Visualize the feature maps for an example image

In [24]:

```
from keras.models import Model
#This will create a model that outputs the feature maps generated the output of block 1
feature_model = Model(inputs=base_model.inputs, outputs=base_model.layers[17].output)
feature_model.summary()
```

```
Model: "functional_3"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
input_1 (InputLayer)         [(None, 224, 224, 3)]     0
_________________________________________________________________
Conv1_pad (ZeroPadding2D)    (None, 225, 225, 3)       0
_________________________________________________________________
Conv1 (Conv2D)               (None, 112, 112, 32)      864
_________________________________________________________________
bn_Conv1 (BatchNormalization (None, 112, 112, 32)      128
_________________________________________________________________
Conv1_relu (ReLU)            (None, 112, 112, 32)      0
_________________________________________________________________
expanded_conv_depthwise (Dep (None, 112, 112, 32)      288
_________________________________________________________________
expanded_conv_depthwise_BN ( (None, 112, 112, 32)      128
_________________________________________________________________
expanded_conv_depthwise_relu (None, 112, 112, 32)      0
_________________________________________________________________
expanded_conv_project (Conv2 (None, 112, 112, 16)      512
_________________________________________________________________
expanded_conv_project_BN (Ba (None, 112, 112, 16)      64
_________________________________________________________________
block_1_expand (Conv2D)      (None, 112, 112, 96)      1536
_________________________________________________________________
block_1_expand_BN (BatchNorm (None, 112, 112, 96)      384
_________________________________________________________________
block_1_expand_relu (ReLU)   (None, 112, 112, 96)      0
_________________________________________________________________
block_1_pad (ZeroPadding2D)  (None, 113, 113, 96)      0
_________________________________________________________________
block_1_depthwise (Depthwise (None, 56, 56, 96)        864
_________________________________________________________________
block_1_depthwise_BN (BatchN (None, 56, 56, 96)        384
_________________________________________________________________
block_1_depthwise_relu (ReLU (None, 56, 56, 96)        0
_________________________________________________________________
block_1_project (Conv2D)     (None, 56, 56, 24)        2304
=================================================================
Total params: 7,456
Trainable params: 0
Non-trainable params: 7,456
_________________________________________________________________
```

In [25]:

```
#Here a single image is loaded in order to generate the feature maps
from keras.preprocessing.image import load_img
from keras.preprocessing.image import img_to_array
from numpy import expand_dims
animal = 'elk'
file = os.listdir(PATH+animal)[3]
#Load a sample image and size resize to the shape expected by the model (img_height, img_wi
dth)
img = load_img(PATH+animal+'/'+file, target_size=(img_height, img_width))
#Convert to an array
img = img_to_array(img)
#Expand dimenstion so that it represents a single 'sample
img = expand_dims(img, axis=0)
#Prepare the imgage for Mobilnet
img = preprocess_input(img)
```

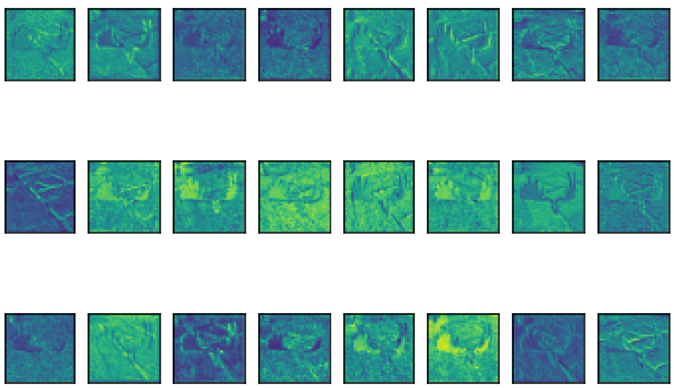
In [26]:

```
#This will process the single image loaded above throught the base model
feature_maps = feature_model.predict(img)
```

It is interesting to note the fine details identified in the first block of MobileNet. Upon changing to be the last output block the feature maps are very generalized.

In [27]:

```
#plot 24 maps on the output of the first block in MobileNet
rows = 3
columns = 8
ix = 1
for _ in range(rows):
    for _ in range(columns):
        ax = plt.subplot(rows, columns, ix)

        ax.set_xticks([])
        ax.set_yticks([])
        plt.imshow(feature_maps[0,:,:,ix-1], cmap='viridis')
        ix += 1
plt.rcParams["figure.figsize"] = (20,20)
plt.show()
```



Fine Tuning

This next section Block 16 of the base model will be set as trainable. The model will them be trained to see if improvements can be made on the model performance.

In [28]:

```
base_model.trainable = True
```

In [29]:

```
num_layers = len(base_model.layers)
print('Number of layers in the base model: ', num_layers)

fine_tune_at = num_layers-16 #this will unfreeze the block15 and block16

for layer in base_model.layers[:fine_tune_at]:
    layer.trainable = False
```

```
Number of layers in the base model:  155
```

Re-compile the model

Since I'm changing the trainable parameter of some layers of the model it needs to be re-compiled before retraining in order that the changes take effect.

In [30]:

```
base_learning_rate = 0.0001
model.compile(optimizer=tf.keras.optimizers.Adam(lr=base_learning_rate/10),
              loss=tf.keras.losses.SparseCategoricalCrossentropy(from_logits=True),
              metrics=['accuracy'])
```

In [31]:

```
model.summary()
```

```
Model: "functional_1"
_________________________________________________________________
Layer (type)                 Output Shape              Param #   
=================================================================
input_2 (InputLayer)         [(None, 224, 224, 3)]     0         
_________________________________________________________________
tf_op_layer_RealDiv (TensorF [(None, 224, 224, 3)]     0         
_________________________________________________________________
tf_op_layer_Sub (TensorFlowO [(None, 224, 224, 3)]     0         
_________________________________________________________________
mobilenetv2_1.00_224 (Functi (None, 7, 7, 1280)        2257984   
_________________________________________________________________
global_average_pooling2d (Gl (None, 1280)              0         
_________________________________________________________________
dropout (Dropout)            (None, 1280)              0         
_________________________________________________________________
prediction (Dense)           (None, 20)                25620     
=================================================================
Total params: 2,283,604
Trainable params: 1,067,540
Non-trainable params: 1,216,064
_________________________________________________________________
```

In [32]:

```
fine_tune_epochs = 20
total_epochs = epochs + fine_tune_epochs
history_fine = model.fit(train_ds,
                    validation_data=val_ds,
                    initial_epoch = history.epoch[-1],
                    epochs=total_epochs)
```

```
Epoch 10/30
351/351 [==============================] - 14s 41ms/step - loss: 2.2912 - accuracy: 0.8
317 - val_loss: 2.2538 - val_accuracy: 0.8569
Epoch 11/30
351/351 [==============================] - 13s 38ms/step - loss: 2.2510 - accuracy: 0.8
659 - val_loss: 2.2391 - val_accuracy: 0.8683
Epoch 12/30
351/351 [==============================] - 13s 38ms/step - loss: 2.2303 - accuracy: 0.8
842 - val_loss: 2.2277 - val_accuracy: 0.8772
Epoch 13/30
351/351 [==============================] - 13s 38ms/step - loss: 2.2138 - accuracy: 0.8
952 - val_loss: 2.2196 - val_accuracy: 0.8847
Epoch 14/30
351/351 [==============================] - 13s 38ms/step - loss: 2.2018 - accuracy: 0.9
057 - val_loss: 2.2114 - val_accuracy: 0.8911
Epoch 15/30
351/351 [==============================] - 13s 38ms/step - loss: 2.1921 - accuracy: 0.9
140 - val_loss: 2.2042 - val_accuracy: 0.8983
Epoch 16/30
351/351 [==============================] - 13s 38ms/step - loss: 2.1822 - accuracy: 0.9
220 - val_loss: 2.1990 - val_accuracy: 0.9036
Epoch 17/30
351/351 [==============================] - 13s 38ms/step - loss: 2.1743 - accuracy: 0.9
286 - val_loss: 2.1938 - val_accuracy: 0.9079
Epoch 18/30
351/351 [==============================] - 13s 38ms/step - loss: 2.1679 - accuracy: 0.9
326 - val_loss: 2.1890 - val_accuracy: 0.9101
Epoch 19/30
351/351 [==============================] - 14s 39ms/step - loss: 2.1610 - accuracy: 0.9
384 - val_loss: 2.1854 - val_accuracy: 0.9104
Epoch 20/30
351/351 [==============================] - 13s 38ms/step - loss: 2.1550 - accuracy: 0.9
423 - val_loss: 2.1818 - val_accuracy: 0.9154
Epoch 21/30
351/351 [==============================] - 13s 38ms/step - loss: 2.1501 - accuracy: 0.9
461 - val_loss: 2.1788 - val_accuracy: 0.9172
Epoch 22/30
351/351 [==============================] - 13s 38ms/step - loss: 2.1456 - accuracy: 0.9
490 - val_loss: 2.1761 - val_accuracy: 0.9197
Epoch 23/30
351/351 [==============================] - 14s 40ms/step - loss: 2.1411 - accuracy: 0.9
520 - val_loss: 2.1735 - val_accuracy: 0.9201
Epoch 24/30
351/351 [==============================] - 13s 38ms/step - loss: 2.1376 - accuracy: 0.9
540 - val_loss: 2.1712 - val_accuracy: 0.9215
```

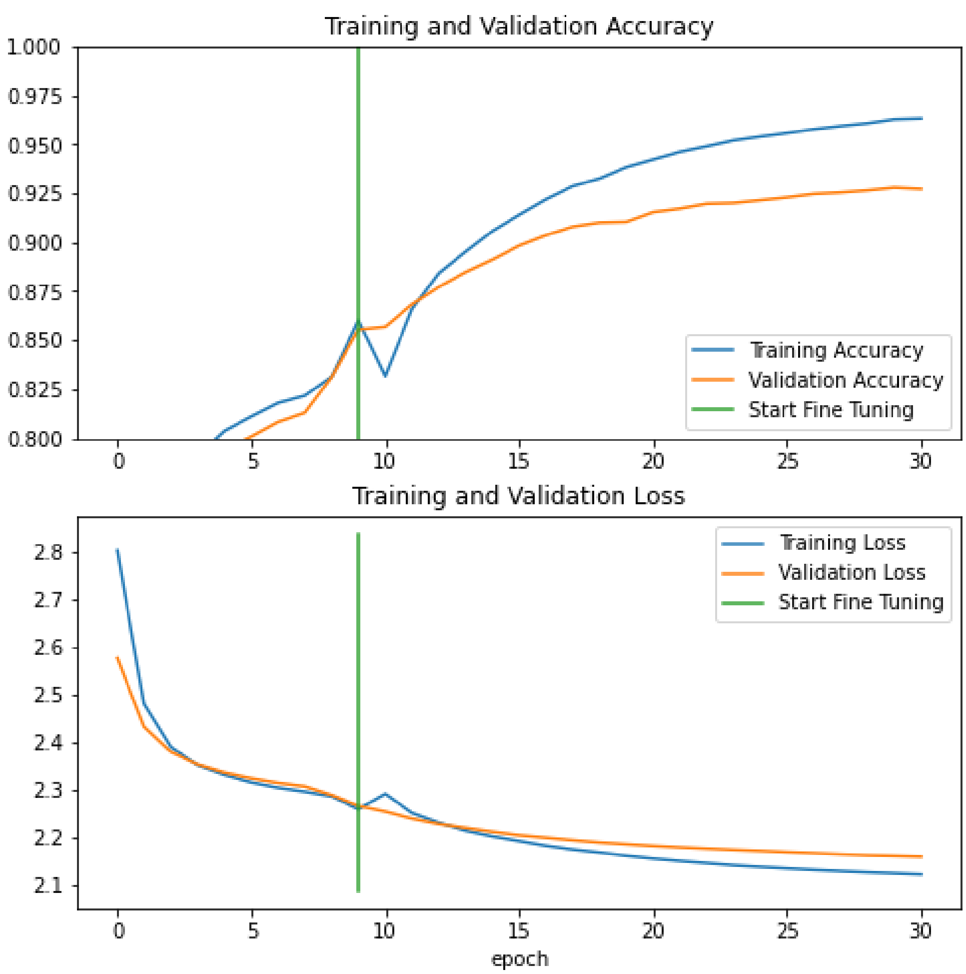
```
Epoch 25/30
351/351 [==============================] - 13s 37ms/step - loss: 2.1346 - accuracy: 0.9
558 - val_loss: 2.1687 - val_accuracy: 0.9229
Epoch 26/30
351/351 [==============================] - 13s 37ms/step - loss: 2.1315 - accuracy: 0.9
577 - val_loss: 2.1665 - val_accuracy: 0.9247
Epoch 27/30
351/351 [==============================] - 13s 37ms/step - loss: 2.1289 - accuracy: 0.9
593 - val_loss: 2.1640 - val_accuracy: 0.9254
Epoch 28/30
351/351 [==============================] - 13s 38ms/step - loss: 2.1264 - accuracy: 0.9
608 - val_loss: 2.1619 - val_accuracy: 0.9265
Epoch 29/30
351/351 [==============================] - 13s 37ms/step - loss: 2.1243 - accuracy: 0.9
628 - val_loss: 2.1607 - val_accuracy: 0.9279
Epoch 30/30
351/351 [==============================] - 13s 37ms/step - loss: 2.1221 - accuracy: 0.9
633 - val_loss: 2.1590 - val_accuracy: 0.9272
```

In [33]:

```
acc += history_fine.history['accuracy']
val_acc += history_fine.history['val_accuracy']

loss += history_fine.history['loss']
val_loss += history_fine.history['val_loss']
```

In [34]:

```
plt.figure(figsize=(8, 8))
plt.subplot(2, 1, 1)
plt.plot(acc, label='Training Accuracy')
plt.plot(val_acc, label='Validation Accuracy')
plt.ylim([0.8, 1])
plt.plot([epochs-1,epochs-1],
          plt.ylim(), label='Start Fine Tuning')
plt.legend(loc='lower right')
plt.title('Training and Validation Accuracy')

plt.subplot(2, 1, 2)
plt.plot(loss, label='Training Loss')
plt.plot(val_loss, label='Validation Loss')
plt.plot([epochs-1,epochs-1],
         plt.ylim(), label='Start Fine Tuning')
plt.legend(loc='upper right')
plt.title('Training and Validation Loss')
plt.xlabel('epoch')
plt.show()
```



In [35]:

```
loss, accuracy = model.evaluate(test_ds)
print('Test dataset loss: ', loss, 'Test dataset accuracy: ', accuracy)
```

```
17/17 [==============================] - 6s 370ms/step - loss: 2.1612 - accuracy: 0.922
8
Test dataset loss:  2.1612000465393066 Test dataset accuracy:  0.9227941036224365
```

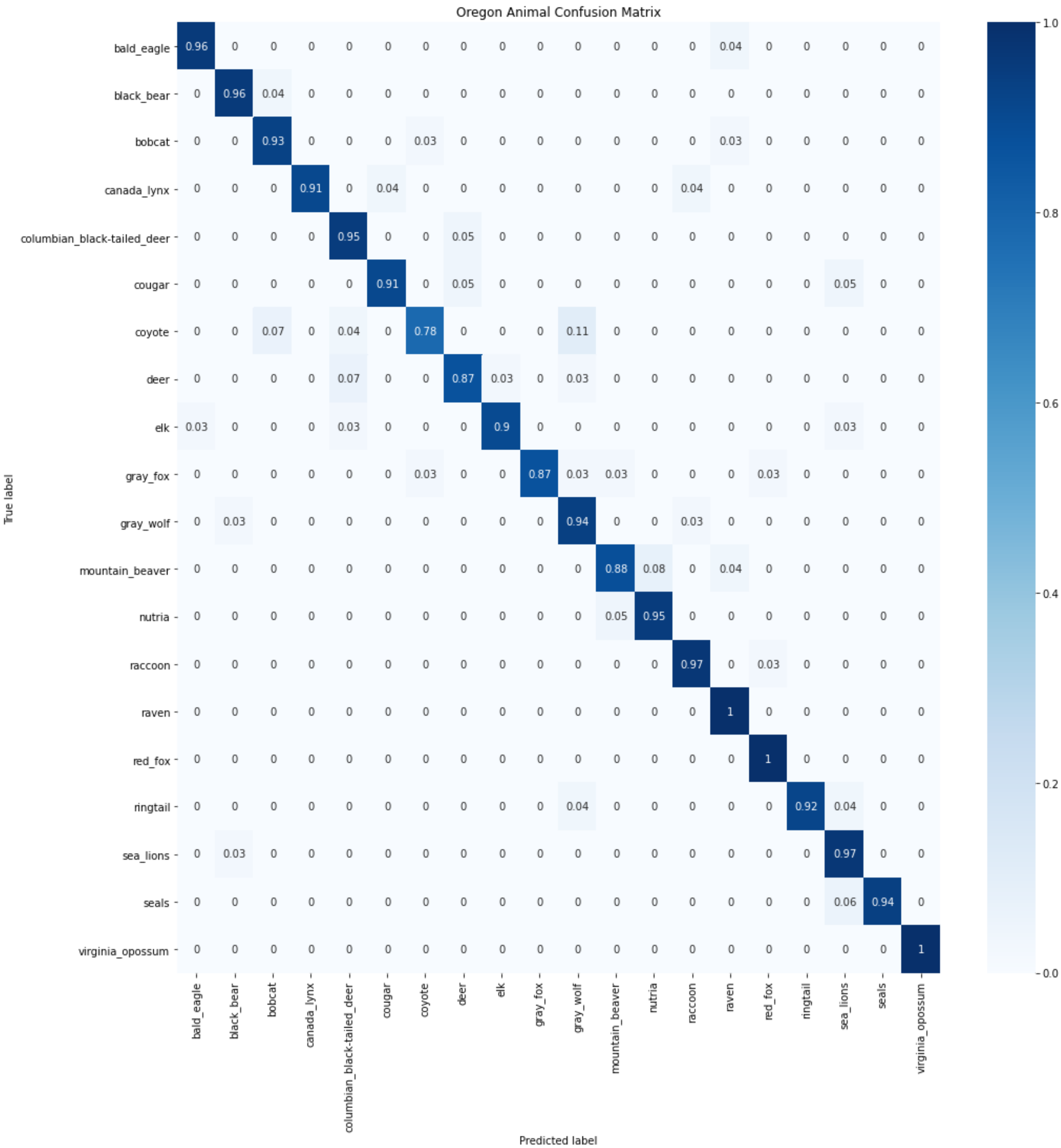
Thanks to the following website for hints on converting the tensorflow confusion matrix to the dataframe and plotting: TensorFlow Keras Confusion Matrix in TensorBoard

In [36]:

```
predictions = np.array([])
labels = np.array([])
for x, y in test_ds:
    predictions = np.concatenate([predictions, np.argmax(model.predict(x), axis=1)])
    labels = np.concatenate([labels, y.numpy()])
    #print('predict= ', predictions, 'label= ', labels)
con_mat = tf.math.confusion_matrix(labels=labels, predictions=predictions).numpy()
con_mat_norm = np.around(con_mat.astype('float')/con_mat.sum(axis=1)[:, np.newaxis], decima
ls=2)
con_mat_df = pd.DataFrame(con_mat_norm, index=class_names, columns=class_names)
```

In [37]:

```
import seaborn as sns
figure = plt.figure(figsize=(15,15))
sns.heatmap(con_mat_df, annot=True, cmap=plt.cm.Blues)
plt.tight_layout()
plt.ylabel('True label')
plt.xlabel('Predicted label')
plt.title('Oregon Animal Confusion Matrix')
plt.show()
```



Save the model and the weights that could be used later.

In [38]:

```
# serialize model to JSON
model_json = model.to_json()
with open("model.json", "w") as json_file:
    json_file.write(model_json)
# serialize weights to HDF5
model.save_weights("model.h5")
```

In [ ]: