

DenkowskiStanislaw4

November 4, 2022

```
[ ]: from jmd_imagescraper.core import *
from jmd_imagescraper.imagecleaner import *
from pathlib import Path
import matplotlib.pyplot as plt
import tensorflow as tf
import numpy as np
import os
```

```
[ ]: root = Path().cwd()/"images"
if not os.path.isdir(root):
    duckduckgo_search(root, 'sushi', ' ', 333)
    duckduckgo_search(root, 'ramen', ' ', 333)
    duckduckgo_search(root, 'taiyaki', ' ', 333)
display_image_cleaner(root)
```

```
HBox(children=(Button(description='|<<', layout=Layout(width='60px'),
↳style=ButtonStyle()), Button(description=...
```

```
HTML(value='<h2>No images left to display in this folder.</h2>',
↳layout=Layout(visibility='hidden'))
```

```
GridBox(children=(VBox(children=(Image(value=b'',
↳layout="Layout(width='150px')"), Button(description='Delete'...
```

```
[ ]: seed = 666

def normali(image, label):
    return image/255.,label

def load_ds(root=root, image_size=(32,32)):
    train_ds, test_ds = tf.keras.preprocessing.image_dataset_from_directory(
        root, label_mode='categorical', batch_size=64, image_size=image_size,
        seed=seed, validation_split=0.2, subset='both')
    return train_ds.map(normali), test_ds.map(normali), train_ds.class_names

train_ds, test_ds, class_names = load_ds()
print(class_names)
```

Found 946 files belonging to 3 classes.
Using 757 files for training.

Using 189 files for validation.
Metal device set to: Apple M1 Pro

systemMemory: 32.00 GB
maxCacheSize: 10.67 GB

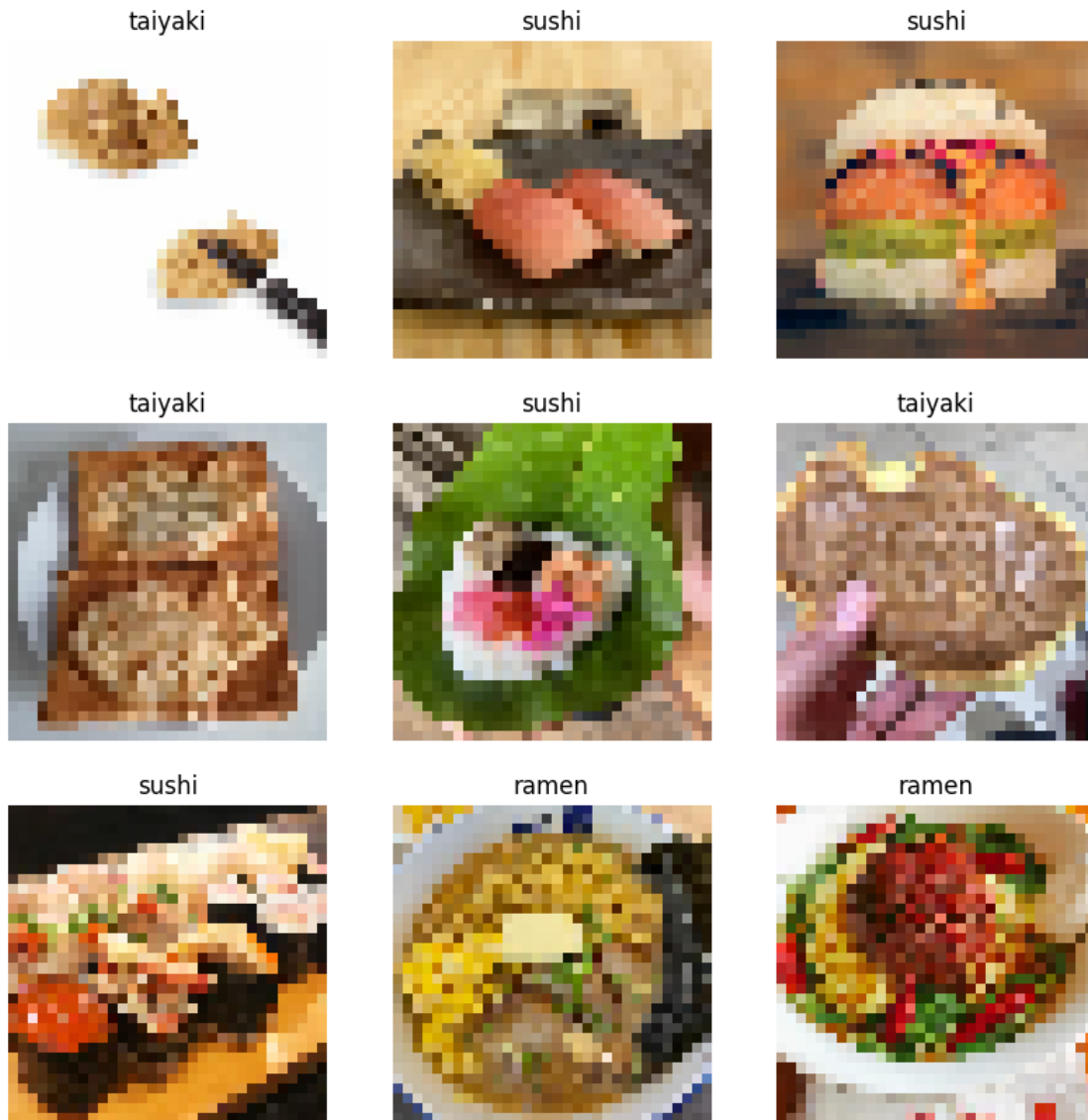
['ramen', 'sushi', 'taiyaki']

2022-11-04 00:22:32.419345: I
tensorflow/core/common_runtime/pluggable_device/pluggable_device_factory.cc:306]
Could not identify NUMA node of platform GPU ID 0, defaulting to 0. Your kernel
may not have been built with NUMA support.
2022-11-04 00:22:32.419493: I
tensorflow/core/common_runtime/pluggable_device/pluggable_device_factory.cc:272]
Created TensorFlow device (/job:localhost/replica:0/task:0/device:GPU:0 with 0
MB memory) -> physical PluggableDevice (device: 0, name: METAL, pci bus id:
<undefined>)

Wykorzystałem `tf.keras.preprocessing.image_dataset_from_dictionary()`, aby wczytać dataset.
Możemy ustawić validation split i jednocześnie wczytać oba zbiory oraz ustawić rozmiar obrazków.

```
[ ]: plt.figure(figsize=(10, 10))  
      for image, label in train_ds.take(1):  
          for i in range(9):  
              ax = plt.subplot(3, 3, i + 1)  
              plt.imshow(image[i])  
              plt.title(class_names[tf.math.argmax(label[i])])  
              plt.axis("off")
```

2022-11-04 00:22:32.575765: W
tensorflow/core/platform/profile_utils/cpu_utils.cc:128] Failed to get CPU
frequency: 0 Hz



1 Własny model

```
[ ]: def block(channels, activation='relu', input_shape=None,
    ↪batch_normalisation=False, dropout=None, gap=False):
    if input_shape:
        conv1 = tf.keras.layers.Conv2D(channels, 3, padding='same',
    ↪activation=activation, input_shape=input_shape)
    else:
        conv1 = tf.keras.layers.Conv2D(channels, 3, padding='same',
    ↪activation=activation)
```

```

conv2 = tf.keras.layers.Conv2D(channels, 3, padding='same',
↪activation=activation)
if batch_normalisation:
    batch_norm1 = tf.keras.layers.BatchNormalization()
    batch_norm2 = tf.keras.layers.BatchNormalization()
if dropout:
    drops = tf.keras.layers.Dropout(dropout)
if gap:
    pool = tf.keras.layers.
↪GlobalAveragePooling2D(data_format='channels_last', keepdims=False)
else:
    pool = tf.keras.layers.MaxPool2D(pool_size=(2,2), padding='valid')

if dropout:
    return [conv1, batch_norm1, conv2, batch_norm2, pool, drops]
elif batch_normalisation:
    return [conv1, batch_norm1, conv2, batch_norm2, pool]
else:
    return [conv1, conv2, pool]

```

```

[ ]: def smooth(y, window=5):
    padded_y = [y[0]]*window+y+[y[-1]]*window
    res = []
    for ix in range(len(y)):
        res.append(np.mean(y[ix:ix+2*window]))
    assert(len(y)==len(res))
    return res

def print_train_and_val(history, gtype):
    smoothy = smooth(history[gtype])
    smoothyval = smooth(history['val_'+gtype])
    plt.plot(smoothy, 'g')
    plt.plot(smoothyval, 'r')
    plt.title('Model '+gtype)
    plt.ylabel(gtype)
    plt.xlabel('epoch')
    plt.legend(['train', 'valid'], loc='upper left')
    plt.show()

def print_history(history):
    print_train_and_val(history, 'accuracy')
    print_train_and_val(history, 'loss')

```

```

[ ]: model = tf.keras.Sequential(
    block(channels=20, activation='relu', input_shape=(None, None, 3),
↪batch_normalisation=True, dropout=0.1) +

```

```

    block(channels=40, activation='relu', batch_normalisation=True, dropout=0.
↪2) +
    block(channels=80, activation='relu', batch_normalisation=True, dropout=0.
↪3) +
    block(channels=160, activation='relu', batch_normalisation=True, dropout=0.
↪4, gap=True) + [
    tf.keras.layers.Dense(units=3,activation='softmax')
])

```

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

Model: "sequential"

Layer (type)	Output Shape	Param #
conv2d (Conv2D)	(None, None, None, 20)	560
batch_normalization (Batch Normalization)	(None, None, None, 20)	80
conv2d_1 (Conv2D)	(None, None, None, 20)	3620
batch_normalization_1 (Batch Normalization)	(None, None, None, 20)	80
max_pooling2d (MaxPooling2D)	(None, None, None, 20)	0
dropout (Dropout)	(None, None, None, 20)	0
conv2d_2 (Conv2D)	(None, None, None, 40)	7240
batch_normalization_2 (Batch Normalization)	(None, None, None, 40)	160
conv2d_3 (Conv2D)	(None, None, None, 40)	14440
batch_normalization_3 (Batch Normalization)	(None, None, None, 40)	160
max_pooling2d_1 (MaxPooling2D)	(None, None, None, 40)	0
dropout_1 (Dropout)	(None, None, None, 40)	0
conv2d_4 (Conv2D)	(None, None, None, 80)	28880

batch_normalization_4 (Batch Normalization)	(None, None, None, 80)	320
conv2d_5 (Conv2D)	(None, None, None, 80)	57680
batch_normalization_5 (Batch Normalization)	(None, None, None, 80)	320
max_pooling2d_2 (MaxPooling2D)	(None, None, None, 80)	0
dropout_2 (Dropout)	(None, None, None, 80)	0
conv2d_6 (Conv2D)	(None, None, None, 160)	115360
batch_normalization_6 (Batch Normalization)	(None, None, None, 160)	640
conv2d_7 (Conv2D)	(None, None, None, 160)	230560
batch_normalization_7 (Batch Normalization)	(None, None, None, 160)	640
global_average_pooling2d (GlobalAveragePooling2D)	(None, 160)	0
dropout_3 (Dropout)	(None, 160)	0
dense (Dense)	(None, 3)	483

```

=====
Total params: 461,223
Trainable params: 460,023
Non-trainable params: 1,200
-----

```

```
[ ]: model.compile(
    optimizer=tf.keras.optimizers.SGD(learning_rate=0.001, momentum=0.9),
    loss=tf.keras.losses.CategoricalCrossentropy(),
    metrics=['accuracy']
)
```

```
[ ]: history = model.fit(
    x=train_ds,
    batch_size=64,
    epochs=150,
    verbose=1,
```

```
validation_data=test_ds
)
```

Epoch 1/150

2022-11-04 00:22:34.109595: I
tensorflow/core/grappler/optimizers/custom_graph_optimizer_registry.cc:114]
Plugin optimizer for device_type GPU is enabled.

12/12 [=====] - ETA: 0s - loss: 1.1009 - accuracy:
0.3989

2022-11-04 00:22:35.275725: I
tensorflow/core/grappler/optimizers/custom_graph_optimizer_registry.cc:114]
Plugin optimizer for device_type GPU is enabled.

12/12 [=====] - 2s 76ms/step - loss: 1.1009 - accuracy:
0.3989 - val_loss: 1.0966 - val_accuracy: 0.3651

Epoch 2/150

12/12 [=====] - 0s 30ms/step - loss: 0.9707 - accuracy:
0.5099 - val_loss: 1.1015 - val_accuracy: 0.3228

Epoch 3/150

12/12 [=====] - 0s 30ms/step - loss: 0.9113 - accuracy:
0.5720 - val_loss: 1.1100 - val_accuracy: 0.3228

Epoch 4/150

12/12 [=====] - 0s 28ms/step - loss: 0.8068 - accuracy:
0.6341 - val_loss: 1.1160 - val_accuracy: 0.3175

Epoch 5/150

12/12 [=====] - 0s 30ms/step - loss: 0.7908 - accuracy:
0.6552 - val_loss: 1.1298 - val_accuracy: 0.4392

Epoch 6/150

12/12 [=====] - 0s 28ms/step - loss: 0.7474 - accuracy:
0.6962 - val_loss: 1.1573 - val_accuracy: 0.3651

Epoch 7/150

12/12 [=====] - 0s 29ms/step - loss: 0.7239 - accuracy:
0.6922 - val_loss: 1.1767 - val_accuracy: 0.3228

Epoch 8/150

12/12 [=====] - 0s 29ms/step - loss: 0.6831 - accuracy:
0.7041 - val_loss: 1.1631 - val_accuracy: 0.3915

Epoch 9/150

12/12 [=====] - 0s 29ms/step - loss: 0.6455 - accuracy:
0.7279 - val_loss: 1.1930 - val_accuracy: 0.3228

Epoch 10/150

12/12 [=====] - 0s 30ms/step - loss: 0.6070 - accuracy:
0.7530 - val_loss: 1.2049 - val_accuracy: 0.3228

Epoch 11/150

12/12 [=====] - 0s 29ms/step - loss: 0.5823 - accuracy:
0.7596 - val_loss: 1.2823 - val_accuracy: 0.3228

Epoch 12/150

12/12 [=====] - 0s 30ms/step - loss: 0.5450 - accuracy:

0.7873 - val_loss: 1.2612 - val_accuracy: 0.3228
 Epoch 13/150
 12/12 [=====] - 0s 32ms/step - loss: 0.5339 - accuracy:
 0.7900 - val_loss: 1.3176 - val_accuracy: 0.3228
 Epoch 14/150
 12/12 [=====] - 0s 32ms/step - loss: 0.5102 - accuracy:
 0.7860 - val_loss: 1.3701 - val_accuracy: 0.3228
 Epoch 15/150
 12/12 [=====] - 0s 28ms/step - loss: 0.4768 - accuracy:
 0.8151 - val_loss: 1.4829 - val_accuracy: 0.3228
 Epoch 16/150
 12/12 [=====] - 0s 29ms/step - loss: 0.4427 - accuracy:
 0.8322 - val_loss: 1.5446 - val_accuracy: 0.3228
 Epoch 17/150
 12/12 [=====] - 0s 30ms/step - loss: 0.4233 - accuracy:
 0.8415 - val_loss: 1.5344 - val_accuracy: 0.3228
 Epoch 18/150
 12/12 [=====] - 0s 29ms/step - loss: 0.3953 - accuracy:
 0.8481 - val_loss: 1.7389 - val_accuracy: 0.3228
 Epoch 19/150
 12/12 [=====] - 0s 31ms/step - loss: 0.3849 - accuracy:
 0.8560 - val_loss: 1.5507 - val_accuracy: 0.3228
 Epoch 20/150
 12/12 [=====] - 0s 30ms/step - loss: 0.3749 - accuracy:
 0.8507 - val_loss: 1.7682 - val_accuracy: 0.3228
 Epoch 21/150
 12/12 [=====] - 0s 30ms/step - loss: 0.3403 - accuracy:
 0.8745 - val_loss: 1.6736 - val_accuracy: 0.3228
 Epoch 22/150
 12/12 [=====] - 0s 30ms/step - loss: 0.3387 - accuracy:
 0.8692 - val_loss: 1.6044 - val_accuracy: 0.3228
 Epoch 23/150
 12/12 [=====] - 0s 30ms/step - loss: 0.3222 - accuracy:
 0.8904 - val_loss: 1.8378 - val_accuracy: 0.3228
 Epoch 24/150
 12/12 [=====] - 0s 30ms/step - loss: 0.2832 - accuracy:
 0.8996 - val_loss: 1.6719 - val_accuracy: 0.3439
 Epoch 25/150
 12/12 [=====] - 0s 32ms/step - loss: 0.2870 - accuracy:
 0.8970 - val_loss: 1.2317 - val_accuracy: 0.4868
 Epoch 26/150
 12/12 [=====] - 0s 28ms/step - loss: 0.2612 - accuracy:
 0.8930 - val_loss: 1.1753 - val_accuracy: 0.5132
 Epoch 27/150
 12/12 [=====] - 0s 31ms/step - loss: 0.2496 - accuracy:
 0.9155 - val_loss: 1.5860 - val_accuracy: 0.4233
 Epoch 28/150
 12/12 [=====] - 0s 29ms/step - loss: 0.2239 - accuracy:

0.9287 - val_loss: 1.4523 - val_accuracy: 0.4180
 Epoch 29/150
 12/12 [=====] - 0s 29ms/step - loss: 0.2404 - accuracy:
 0.9036 - val_loss: 1.8575 - val_accuracy: 0.4233
 Epoch 30/150
 12/12 [=====] - 1s 32ms/step - loss: 0.2228 - accuracy:
 0.9155 - val_loss: 1.1708 - val_accuracy: 0.5185
 Epoch 31/150
 12/12 [=====] - 1s 32ms/step - loss: 0.2344 - accuracy:
 0.9141 - val_loss: 1.1973 - val_accuracy: 0.5291
 Epoch 32/150
 12/12 [=====] - 1s 31ms/step - loss: 0.1926 - accuracy:
 0.9273 - val_loss: 1.3032 - val_accuracy: 0.5344
 Epoch 33/150
 12/12 [=====] - 0s 29ms/step - loss: 0.2099 - accuracy:
 0.9168 - val_loss: 1.5164 - val_accuracy: 0.5450
 Epoch 34/150
 12/12 [=====] - 0s 29ms/step - loss: 0.1878 - accuracy:
 0.9260 - val_loss: 1.6810 - val_accuracy: 0.4921
 Epoch 35/150
 12/12 [=====] - 0s 29ms/step - loss: 0.1813 - accuracy:
 0.9406 - val_loss: 0.7089 - val_accuracy: 0.7143
 Epoch 36/150
 12/12 [=====] - 0s 29ms/step - loss: 0.1544 - accuracy:
 0.9511 - val_loss: 1.2402 - val_accuracy: 0.5714
 Epoch 37/150
 12/12 [=====] - 0s 29ms/step - loss: 0.1481 - accuracy:
 0.9511 - val_loss: 0.7618 - val_accuracy: 0.7037
 Epoch 38/150
 12/12 [=====] - 1s 33ms/step - loss: 0.1563 - accuracy:
 0.9406 - val_loss: 1.3356 - val_accuracy: 0.5450
 Epoch 39/150
 12/12 [=====] - 1s 32ms/step - loss: 0.1747 - accuracy:
 0.9366 - val_loss: 0.5711 - val_accuracy: 0.7672
 Epoch 40/150
 12/12 [=====] - 0s 30ms/step - loss: 0.1709 - accuracy:
 0.9339 - val_loss: 1.4390 - val_accuracy: 0.5397
 Epoch 41/150
 12/12 [=====] - 0s 29ms/step - loss: 0.1293 - accuracy:
 0.9524 - val_loss: 0.7042 - val_accuracy: 0.7513
 Epoch 42/150
 12/12 [=====] - 0s 28ms/step - loss: 0.1240 - accuracy:
 0.9617 - val_loss: 0.5718 - val_accuracy: 0.7989
 Epoch 43/150
 12/12 [=====] - 0s 28ms/step - loss: 0.1266 - accuracy:
 0.9630 - val_loss: 0.6355 - val_accuracy: 0.7831
 Epoch 44/150
 12/12 [=====] - 0s 28ms/step - loss: 0.1207 - accuracy:

0.9657 - val_loss: 0.7296 - val_accuracy: 0.7672
 Epoch 45/150
 12/12 [=====] - 0s 30ms/step - loss: 0.1275 - accuracy:
 0.9617 - val_loss: 1.0255 - val_accuracy: 0.6667
 Epoch 46/150
 12/12 [=====] - 0s 29ms/step - loss: 0.1014 - accuracy:
 0.9657 - val_loss: 0.6551 - val_accuracy: 0.7672
 Epoch 47/150
 12/12 [=====] - 0s 30ms/step - loss: 0.0979 - accuracy:
 0.9683 - val_loss: 0.7580 - val_accuracy: 0.7513
 Epoch 48/150
 12/12 [=====] - 0s 28ms/step - loss: 0.0873 - accuracy:
 0.9789 - val_loss: 0.5137 - val_accuracy: 0.8466
 Epoch 49/150
 12/12 [=====] - 0s 29ms/step - loss: 0.0917 - accuracy:
 0.9749 - val_loss: 0.4611 - val_accuracy: 0.8466
 Epoch 50/150
 12/12 [=====] - 0s 28ms/step - loss: 0.1099 - accuracy:
 0.9604 - val_loss: 0.7865 - val_accuracy: 0.7566
 Epoch 51/150
 12/12 [=====] - 0s 30ms/step - loss: 0.0812 - accuracy:
 0.9789 - val_loss: 0.8607 - val_accuracy: 0.7249
 Epoch 52/150
 12/12 [=====] - 0s 28ms/step - loss: 0.0823 - accuracy:
 0.9723 - val_loss: 0.6842 - val_accuracy: 0.7831
 Epoch 53/150
 12/12 [=====] - 0s 30ms/step - loss: 0.0691 - accuracy:
 0.9828 - val_loss: 0.5947 - val_accuracy: 0.8201
 Epoch 54/150
 12/12 [=====] - 0s 29ms/step - loss: 0.0729 - accuracy:
 0.9789 - val_loss: 0.5361 - val_accuracy: 0.8413
 Epoch 55/150
 12/12 [=====] - 0s 28ms/step - loss: 0.0836 - accuracy:
 0.9775 - val_loss: 1.2213 - val_accuracy: 0.6931
 Epoch 56/150
 12/12 [=====] - 0s 29ms/step - loss: 0.0700 - accuracy:
 0.9762 - val_loss: 0.5492 - val_accuracy: 0.8307
 Epoch 57/150
 12/12 [=====] - 0s 28ms/step - loss: 0.0690 - accuracy:
 0.9789 - val_loss: 0.4581 - val_accuracy: 0.8730
 Epoch 58/150
 12/12 [=====] - 0s 29ms/step - loss: 0.0613 - accuracy:
 0.9828 - val_loss: 0.4581 - val_accuracy: 0.8677
 Epoch 59/150
 12/12 [=====] - 0s 28ms/step - loss: 0.0563 - accuracy:
 0.9855 - val_loss: 0.9831 - val_accuracy: 0.7513
 Epoch 60/150
 12/12 [=====] - 0s 28ms/step - loss: 0.0794 - accuracy:

0.9749 - val_loss: 0.8149 - val_accuracy: 0.7619
 Epoch 61/150
 12/12 [=====] - 0s 29ms/step - loss: 0.0639 - accuracy:
 0.9815 - val_loss: 1.1705 - val_accuracy: 0.6772
 Epoch 62/150
 12/12 [=====] - 0s 31ms/step - loss: 0.0614 - accuracy:
 0.9815 - val_loss: 0.4431 - val_accuracy: 0.8413
 Epoch 63/150
 12/12 [=====] - 0s 27ms/step - loss: 0.0738 - accuracy:
 0.9749 - val_loss: 0.7429 - val_accuracy: 0.7778
 Epoch 64/150
 12/12 [=====] - 0s 28ms/step - loss: 0.0561 - accuracy:
 0.9841 - val_loss: 0.5229 - val_accuracy: 0.8519
 Epoch 65/150
 12/12 [=====] - 0s 28ms/step - loss: 0.0471 - accuracy:
 0.9894 - val_loss: 0.4712 - val_accuracy: 0.8783
 Epoch 66/150
 12/12 [=====] - 0s 28ms/step - loss: 0.0562 - accuracy:
 0.9815 - val_loss: 0.5833 - val_accuracy: 0.8360
 Epoch 67/150
 12/12 [=====] - 0s 28ms/step - loss: 0.0641 - accuracy:
 0.9815 - val_loss: 0.6504 - val_accuracy: 0.8360
 Epoch 68/150
 12/12 [=====] - 0s 27ms/step - loss: 0.0397 - accuracy:
 0.9894 - val_loss: 0.5127 - val_accuracy: 0.8519
 Epoch 69/150
 12/12 [=====] - 0s 27ms/step - loss: 0.0441 - accuracy:
 0.9868 - val_loss: 0.4903 - val_accuracy: 0.8677
 Epoch 70/150
 12/12 [=====] - 0s 28ms/step - loss: 0.0441 - accuracy:
 0.9934 - val_loss: 0.5784 - val_accuracy: 0.8254
 Epoch 71/150
 12/12 [=====] - 0s 28ms/step - loss: 0.0425 - accuracy:
 0.9881 - val_loss: 0.7996 - val_accuracy: 0.7513
 Epoch 72/150
 12/12 [=====] - 0s 28ms/step - loss: 0.0517 - accuracy:
 0.9868 - val_loss: 0.4978 - val_accuracy: 0.8466
 Epoch 73/150
 12/12 [=====] - 0s 29ms/step - loss: 0.0429 - accuracy:
 0.9921 - val_loss: 1.1785 - val_accuracy: 0.7249
 Epoch 74/150
 12/12 [=====] - 0s 28ms/step - loss: 0.0544 - accuracy:
 0.9828 - val_loss: 0.5861 - val_accuracy: 0.8466
 Epoch 75/150
 12/12 [=====] - 0s 28ms/step - loss: 0.0406 - accuracy:
 0.9881 - val_loss: 0.7347 - val_accuracy: 0.7884
 Epoch 76/150
 12/12 [=====] - 0s 29ms/step - loss: 0.0391 - accuracy:

0.9921 - val_loss: 0.5090 - val_accuracy: 0.8201
 Epoch 77/150
 12/12 [=====] - 0s 29ms/step - loss: 0.0442 - accuracy:
 0.9828 - val_loss: 0.4250 - val_accuracy: 0.8836
 Epoch 78/150
 12/12 [=====] - 0s 29ms/step - loss: 0.0357 - accuracy:
 0.9881 - val_loss: 0.7768 - val_accuracy: 0.7989
 Epoch 79/150
 12/12 [=====] - 0s 28ms/step - loss: 0.0334 - accuracy:
 0.9921 - val_loss: 0.4883 - val_accuracy: 0.8677
 Epoch 80/150
 12/12 [=====] - 0s 28ms/step - loss: 0.0393 - accuracy:
 0.9908 - val_loss: 0.4887 - val_accuracy: 0.8677
 Epoch 81/150
 12/12 [=====] - 0s 31ms/step - loss: 0.0384 - accuracy:
 0.9868 - val_loss: 0.4791 - val_accuracy: 0.8571
 Epoch 82/150
 12/12 [=====] - 0s 29ms/step - loss: 0.0336 - accuracy:
 0.9908 - val_loss: 0.4759 - val_accuracy: 0.8466
 Epoch 83/150
 12/12 [=====] - 0s 30ms/step - loss: 0.0353 - accuracy:
 0.9934 - val_loss: 0.7238 - val_accuracy: 0.7937
 Epoch 84/150
 12/12 [=====] - 0s 29ms/step - loss: 0.0370 - accuracy:
 0.9908 - val_loss: 0.5818 - val_accuracy: 0.8519
 Epoch 85/150
 12/12 [=====] - 1s 36ms/step - loss: 0.0370 - accuracy:
 0.9881 - val_loss: 0.5610 - val_accuracy: 0.8413
 Epoch 86/150
 12/12 [=====] - 1s 31ms/step - loss: 0.0316 - accuracy:
 0.9934 - val_loss: 0.5276 - val_accuracy: 0.8571
 Epoch 87/150
 12/12 [=====] - 0s 30ms/step - loss: 0.0303 - accuracy:
 0.9908 - val_loss: 0.5595 - val_accuracy: 0.8677
 Epoch 88/150
 12/12 [=====] - 0s 29ms/step - loss: 0.0385 - accuracy:
 0.9921 - val_loss: 0.5158 - val_accuracy: 0.8624
 Epoch 89/150
 12/12 [=====] - 0s 29ms/step - loss: 0.0443 - accuracy:
 0.9855 - val_loss: 0.5035 - val_accuracy: 0.8466
 Epoch 90/150
 12/12 [=====] - 0s 31ms/step - loss: 0.0251 - accuracy:
 0.9947 - val_loss: 0.4669 - val_accuracy: 0.8571
 Epoch 91/150
 12/12 [=====] - 1s 33ms/step - loss: 0.0263 - accuracy:
 0.9908 - val_loss: 0.5227 - val_accuracy: 0.8624
 Epoch 92/150
 12/12 [=====] - 1s 32ms/step - loss: 0.0246 - accuracy:

0.9921 - val_loss: 0.5232 - val_accuracy: 0.8571
 Epoch 93/150
 12/12 [=====] - 0s 31ms/step - loss: 0.0237 - accuracy:
 0.9987 - val_loss: 0.8071 - val_accuracy: 0.7937
 Epoch 94/150
 12/12 [=====] - 0s 28ms/step - loss: 0.0314 - accuracy:
 0.9921 - val_loss: 0.6320 - val_accuracy: 0.8360
 Epoch 95/150
 12/12 [=====] - 0s 29ms/step - loss: 0.0229 - accuracy:
 0.9960 - val_loss: 0.5973 - val_accuracy: 0.8571
 Epoch 96/150
 12/12 [=====] - 1s 33ms/step - loss: 0.0207 - accuracy:
 0.9960 - val_loss: 0.5312 - val_accuracy: 0.8519
 Epoch 97/150
 12/12 [=====] - 1s 33ms/step - loss: 0.0232 - accuracy:
 0.9934 - val_loss: 0.6864 - val_accuracy: 0.8148
 Epoch 98/150
 12/12 [=====] - 1s 31ms/step - loss: 0.0283 - accuracy:
 0.9934 - val_loss: 0.8866 - val_accuracy: 0.7619
 Epoch 99/150
 12/12 [=====] - 1s 35ms/step - loss: 0.0259 - accuracy:
 0.9947 - val_loss: 0.6024 - val_accuracy: 0.8571
 Epoch 100/150
 12/12 [=====] - 1s 33ms/step - loss: 0.0190 - accuracy:
 0.9974 - val_loss: 0.7722 - val_accuracy: 0.8307
 Epoch 101/150
 12/12 [=====] - 0s 31ms/step - loss: 0.0201 - accuracy:
 0.9960 - val_loss: 0.5030 - val_accuracy: 0.8571
 Epoch 102/150
 12/12 [=====] - 0s 31ms/step - loss: 0.0191 - accuracy:
 0.9960 - val_loss: 0.5810 - val_accuracy: 0.8360
 Epoch 103/150
 12/12 [=====] - 0s 29ms/step - loss: 0.0196 - accuracy:
 0.9960 - val_loss: 0.6035 - val_accuracy: 0.8624
 Epoch 104/150
 12/12 [=====] - 1s 32ms/step - loss: 0.0187 - accuracy:
 0.9974 - val_loss: 0.4985 - val_accuracy: 0.8889
 Epoch 105/150
 12/12 [=====] - 1s 33ms/step - loss: 0.0204 - accuracy:
 0.9960 - val_loss: 0.4979 - val_accuracy: 0.8677
 Epoch 106/150
 12/12 [=====] - 1s 31ms/step - loss: 0.0227 - accuracy:
 0.9921 - val_loss: 0.4853 - val_accuracy: 0.8730
 Epoch 107/150
 12/12 [=====] - 1s 34ms/step - loss: 0.0128 - accuracy:
 1.0000 - val_loss: 0.5703 - val_accuracy: 0.8413
 Epoch 108/150
 12/12 [=====] - 1s 31ms/step - loss: 0.0200 - accuracy:

0.9960 - val_loss: 0.4549 - val_accuracy: 0.8836
Epoch 109/150
12/12 [=====] - 1s 32ms/step - loss: 0.0342 - accuracy:
0.9881 - val_loss: 0.6247 - val_accuracy: 0.8519
Epoch 110/150
12/12 [=====] - 1s 33ms/step - loss: 0.0207 - accuracy:
0.9921 - val_loss: 0.6083 - val_accuracy: 0.8201
Epoch 111/150
12/12 [=====] - 0s 29ms/step - loss: 0.0238 - accuracy:
0.9934 - val_loss: 0.6565 - val_accuracy: 0.8360
Epoch 112/150
12/12 [=====] - 1s 32ms/step - loss: 0.0243 - accuracy:
0.9934 - val_loss: 0.6724 - val_accuracy: 0.7989
Epoch 113/150
12/12 [=====] - 0s 30ms/step - loss: 0.0218 - accuracy:
0.9934 - val_loss: 0.6060 - val_accuracy: 0.8413
Epoch 114/150
12/12 [=====] - 1s 31ms/step - loss: 0.0177 - accuracy:
0.9960 - val_loss: 0.5766 - val_accuracy: 0.8624
Epoch 115/150
12/12 [=====] - 1s 32ms/step - loss: 0.0193 - accuracy:
0.9987 - val_loss: 1.2542 - val_accuracy: 0.7196
Epoch 116/150
12/12 [=====] - 1s 32ms/step - loss: 0.0165 - accuracy:
0.9960 - val_loss: 0.6663 - val_accuracy: 0.8466
Epoch 117/150
12/12 [=====] - 0s 31ms/step - loss: 0.0164 - accuracy:
0.9960 - val_loss: 0.4968 - val_accuracy: 0.8624
Epoch 118/150
12/12 [=====] - 0s 29ms/step - loss: 0.0328 - accuracy:
0.9855 - val_loss: 0.4959 - val_accuracy: 0.8783
Epoch 119/150
12/12 [=====] - 1s 33ms/step - loss: 0.0179 - accuracy:
0.9960 - val_loss: 0.5482 - val_accuracy: 0.8730
Epoch 120/150
12/12 [=====] - 1s 32ms/step - loss: 0.0234 - accuracy:
0.9934 - val_loss: 0.6242 - val_accuracy: 0.8413
Epoch 121/150
12/12 [=====] - 1s 31ms/step - loss: 0.0156 - accuracy:
0.9974 - val_loss: 0.5818 - val_accuracy: 0.8307
Epoch 122/150
12/12 [=====] - 1s 32ms/step - loss: 0.0175 - accuracy:
0.9960 - val_loss: 0.5226 - val_accuracy: 0.8624
Epoch 123/150
12/12 [=====] - 1s 30ms/step - loss: 0.0124 - accuracy:
0.9987 - val_loss: 0.4803 - val_accuracy: 0.8783
Epoch 124/150
12/12 [=====] - 1s 32ms/step - loss: 0.0159 - accuracy:

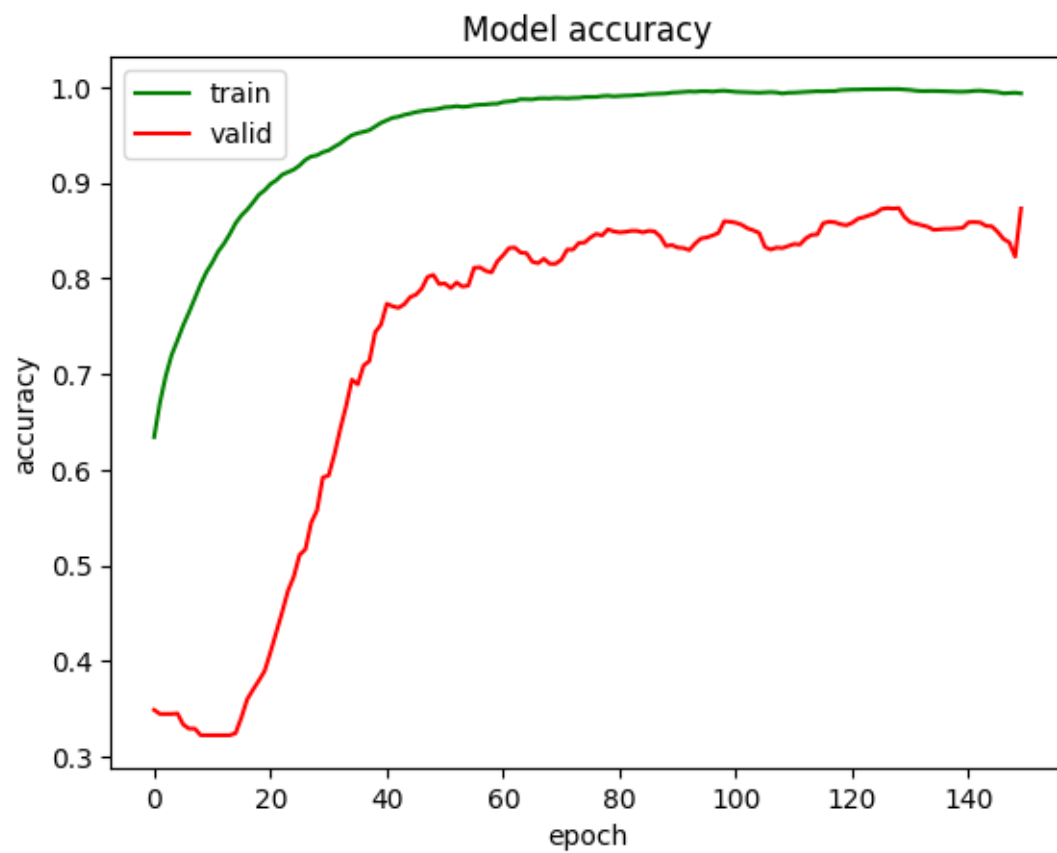
0.9987 - val_loss: 0.4898 - val_accuracy: 0.8677
 Epoch 125/150
 12/12 [=====] - 0s 28ms/step - loss: 0.0154 - accuracy:
 0.9974 - val_loss: 0.6391 - val_accuracy: 0.8360
 Epoch 126/150
 12/12 [=====] - 0s 31ms/step - loss: 0.0135 - accuracy:
 0.9987 - val_loss: 0.5203 - val_accuracy: 0.8624
 Epoch 127/150
 12/12 [=====] - 1s 32ms/step - loss: 0.0140 - accuracy:
 0.9960 - val_loss: 0.6131 - val_accuracy: 0.8571
 Epoch 128/150
 12/12 [=====] - 1s 34ms/step - loss: 0.0177 - accuracy:
 0.9960 - val_loss: 0.5222 - val_accuracy: 0.8571
 Epoch 129/150
 12/12 [=====] - 1s 32ms/step - loss: 0.0134 - accuracy:
 0.9987 - val_loss: 0.5622 - val_accuracy: 0.8624
 Epoch 130/150
 12/12 [=====] - 1s 31ms/step - loss: 0.0137 - accuracy:
 0.9960 - val_loss: 0.5252 - val_accuracy: 0.8677
 Epoch 131/150
 12/12 [=====] - 0s 31ms/step - loss: 0.0136 - accuracy:
 0.9974 - val_loss: 0.5053 - val_accuracy: 0.8730
 Epoch 132/150
 12/12 [=====] - 0s 31ms/step - loss: 0.0104 - accuracy:
 0.9987 - val_loss: 0.4543 - val_accuracy: 0.8783
 Epoch 133/150
 12/12 [=====] - 1s 33ms/step - loss: 0.0106 - accuracy:
 0.9987 - val_loss: 0.4593 - val_accuracy: 0.8995
 Epoch 134/150
 12/12 [=====] - 1s 31ms/step - loss: 0.0084 - accuracy:
 0.9987 - val_loss: 0.4675 - val_accuracy: 0.8889
 Epoch 135/150
 12/12 [=====] - 1s 30ms/step - loss: 0.0108 - accuracy:
 0.9987 - val_loss: 0.5301 - val_accuracy: 0.8783
 Epoch 136/150
 12/12 [=====] - 1s 34ms/step - loss: 0.0091 - accuracy:
 1.0000 - val_loss: 0.5493 - val_accuracy: 0.8730
 Epoch 137/150
 12/12 [=====] - 1s 34ms/step - loss: 0.0140 - accuracy:
 0.9960 - val_loss: 0.5010 - val_accuracy: 0.8519
 Epoch 138/150
 12/12 [=====] - 1s 32ms/step - loss: 0.0167 - accuracy:
 0.9960 - val_loss: 0.4949 - val_accuracy: 0.8624
 Epoch 139/150
 12/12 [=====] - 1s 34ms/step - loss: 0.0235 - accuracy:
 0.9947 - val_loss: 0.8940 - val_accuracy: 0.7672
 Epoch 140/150
 12/12 [=====] - 1s 33ms/step - loss: 0.0237 - accuracy:

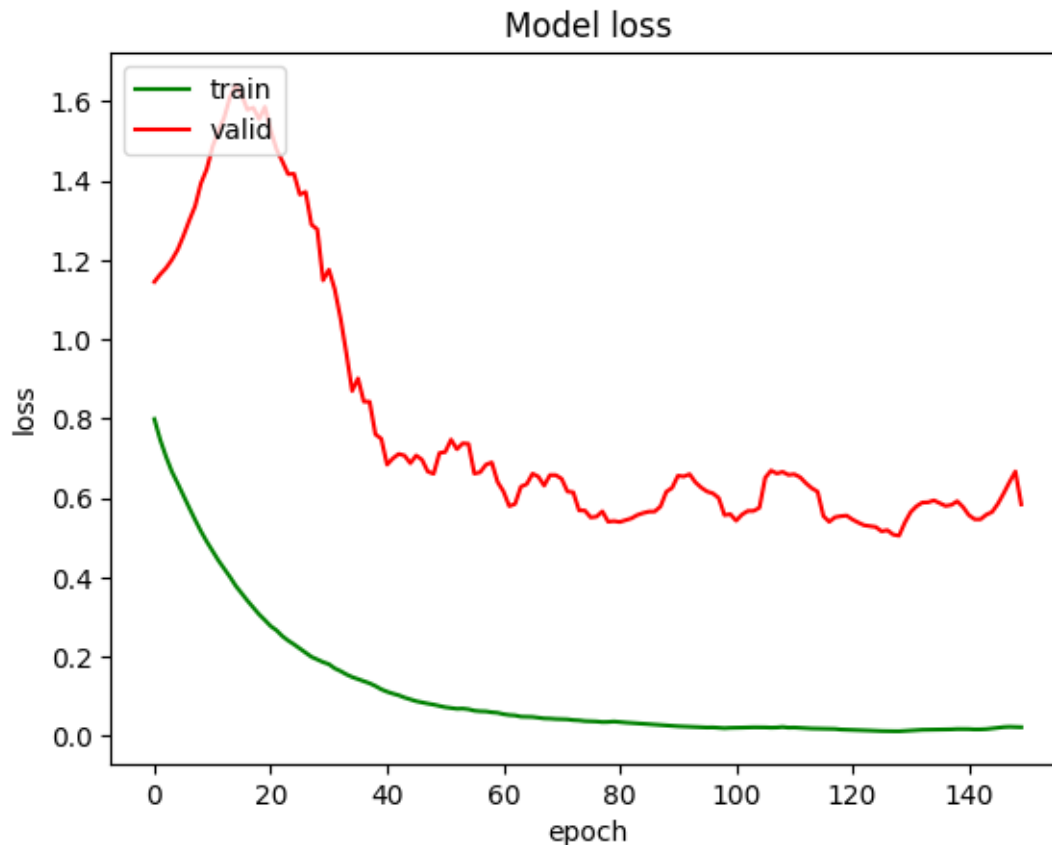
```

0.9908 - val_loss: 0.7894 - val_accuracy: 0.8148
Epoch 141/150
12/12 [=====] - 1s 33ms/step - loss: 0.0242 - accuracy:
0.9894 - val_loss: 0.6517 - val_accuracy: 0.8571
Epoch 142/150
12/12 [=====] - 1s 33ms/step - loss: 0.0158 - accuracy:
0.9947 - val_loss: 0.5467 - val_accuracy: 0.8624
Epoch 143/150
12/12 [=====] - 0s 31ms/step - loss: 0.0119 - accuracy:
1.0000 - val_loss: 0.4656 - val_accuracy: 0.8836
Epoch 144/150
12/12 [=====] - 1s 32ms/step - loss: 0.0076 - accuracy:
0.9987 - val_loss: 0.5159 - val_accuracy: 0.8571
Epoch 145/150
12/12 [=====] - 1s 33ms/step - loss: 0.0122 - accuracy:
0.9974 - val_loss: 0.4607 - val_accuracy: 0.8836
Epoch 146/150
12/12 [=====] - 1s 33ms/step - loss: 0.0121 - accuracy:
1.0000 - val_loss: 0.4796 - val_accuracy: 0.8783
Epoch 147/150
12/12 [=====] - 1s 33ms/step - loss: 0.0203 - accuracy:
0.9921 - val_loss: 0.5268 - val_accuracy: 0.8519
Epoch 148/150
12/12 [=====] - 1s 33ms/step - loss: 0.0241 - accuracy:
0.9934 - val_loss: 0.5878 - val_accuracy: 0.8677
Epoch 149/150
12/12 [=====] - 1s 34ms/step - loss: 0.0234 - accuracy:
0.9947 - val_loss: 0.7494 - val_accuracy: 0.7725
Epoch 150/150
12/12 [=====] - 1s 32ms/step - loss: 0.0222 - accuracy:
0.9934 - val_loss: 0.5841 - val_accuracy: 0.8730

```

```
[ ]: print_history(history.history)
```



Całkiem niezłe wyniki da się osiągnąć, ale jednak wydaje się być trochę ograniczone małymi zbiorami danych. Szczególnie podejrzanie wyglądają duże skoki wyników na zbiorze walidacyjnym.

2 Ładujemy zapisane wcześniej wagi w modelu.

```
[ ]: model = tf.keras.Sequential(
    block(channels=20, activation='relu', input_shape=(None, None, 3),
    ↪batch_normalisation=True, dropout=0.1) +
    block(channels=40, activation='relu', batch_normalisation=True, dropout=0.
    ↪2) +
    block(channels=80, activation='relu', batch_normalisation=True, dropout=0.
    ↪3) +
    block(channels=160, activation='relu', batch_normalisation=True, dropout=0.
    ↪4, gap=True) + [
    tf.keras.layers.Dense(units=10, activation='softmax')
])
```

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

Model: "sequential_1"

Layer (type)	Output Shape	Param #
conv2d_8 (Conv2D)	(None, None, None, 20)	560
batch_normalization_8 (Batch Normalization)	(None, None, None, 20)	80
conv2d_9 (Conv2D)	(None, None, None, 20)	3620
batch_normalization_9 (Batch Normalization)	(None, None, None, 20)	80
max_pooling2d_3 (MaxPooling2D)	(None, None, None, 20)	0
dropout_4 (Dropout)	(None, None, None, 20)	0
conv2d_10 (Conv2D)	(None, None, None, 40)	7240
batch_normalization_10 (Batch Normalization)	(None, None, None, 40)	160
conv2d_11 (Conv2D)	(None, None, None, 40)	14440
batch_normalization_11 (Batch Normalization)	(None, None, None, 40)	160
max_pooling2d_4 (MaxPooling2D)	(None, None, None, 40)	0
dropout_5 (Dropout)	(None, None, None, 40)	0
conv2d_12 (Conv2D)	(None, None, None, 80)	28880
batch_normalization_12 (Batch Normalization)	(None, None, None, 80)	320
conv2d_13 (Conv2D)	(None, None, None, 80)	57680
batch_normalization_13 (Batch Normalization)	(None, None, None, 80)	320
max_pooling2d_5 (MaxPooling2D)	(None, None, None, 80)	0
dropout_6 (Dropout)	(None, None, None, 80)	0

conv2d_14 (Conv2D)	(None, None, None, 160)	115360
batch_normalization_14 (Batch Normalization)	(None, None, None, 160)	640
conv2d_15 (Conv2D)	(None, None, None, 160)	230560
batch_normalization_15 (Batch Normalization)	(None, None, None, 160)	640
global_average_pooling2d_1 (Global Average Pooling2D)	(None, 160)	0
dropout_7 (Dropout)	(None, 160)	0
dense_1 (Dense)	(None, 10)	1610

```

=====
Total params: 462,350
Trainable params: 461,150
Non-trainable params: 1,200
-----

```

```
[ ]: model.load_weights('./checkpoints/CIFAR')
```

```
[ ]: <tensorflow.python.checkpoint.checkpoint.CheckpointLoadStatus at 0x2d2dcbd30>
```

```
[ ]: model.trainable = False
old_model = model.layers[:-1]
old_weights = []
for layer in old_model:
    assert not layer.trainable
    old_weights.append(layer.weights)
```

Ustawiając parametr trainable w modelu, rekurencyjnie wszystkie warstwy mają ustawiane trainable na tą samą wartość.

Dodatkowo korzystając z atrybutu layers mamy dostęp do warstw modelu, i w ten sposób możemy zapomnieć o ostatniej warstwie odpowiadającej za kategoryzację.

```
[ ]: model = tf.keras.Sequential(old_model+[tf.keras.layers.Dense(units=3,
    ↪activation='softmax'))]
```

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

```
Model: "sequential_2"
```

Layer (type)	Output Shape	Param #
conv2d_8 (Conv2D)	(None, None, None, 20)	560

batch_normalization_8 (Batch Normalization)	(None, None, None, 20)	80
conv2d_9 (Conv2D)	(None, None, None, 20)	3620
batch_normalization_9 (Batch Normalization)	(None, None, None, 20)	80
max_pooling2d_3 (MaxPooling2D)	(None, None, None, 20)	0
dropout_4 (Dropout)	(None, None, None, 20)	0
conv2d_10 (Conv2D)	(None, None, None, 40)	7240
batch_normalization_10 (Batch Normalization)	(None, None, None, 40)	160
conv2d_11 (Conv2D)	(None, None, None, 40)	14440
batch_normalization_11 (Batch Normalization)	(None, None, None, 40)	160
max_pooling2d_4 (MaxPooling2D)	(None, None, None, 40)	0
dropout_5 (Dropout)	(None, None, None, 40)	0
conv2d_12 (Conv2D)	(None, None, None, 80)	28880
batch_normalization_12 (Batch Normalization)	(None, None, None, 80)	320
conv2d_13 (Conv2D)	(None, None, None, 80)	57680
batch_normalization_13 (Batch Normalization)	(None, None, None, 80)	320
max_pooling2d_5 (MaxPooling2D)	(None, None, None, 80)	0
dropout_6 (Dropout)	(None, None, None, 80)	0
conv2d_14 (Conv2D)	(None, None, None, 160)	115360
batch_normalization_14 (Batch Normalization)	(None, None, None, 160)	640

conv2d_15 (Conv2D)	(None, None, None, 160)	230560
batch_normalization_15 (Batch Normalization)	(None, None, None, 160)	640
global_average_pooling2d_1 (GlobalAveragePooling2D)	(None, 160)	0
dropout_7 (Dropout)	(None, 160)	0
dense_2 (Dense)	(None, 3)	483

```

=====
Total params: 461,223
Trainable params: 483
Non-trainable params: 460,740
-----

```

```
[ ]: for layer in model.layers[:-1]:
      assert not layer.trainable
      assert model.layers[-1].trainable
```

Dodałem ostatnią warstwę wyjściową z 3 kategoriami. Odpowiednio ustawiłem trainable aby upewnić się, że warstwa będzie się uczyła.

```
[ ]: model.compile(
      optimizer=tf.keras.optimizers.SGD(learning_rate=0.001, momentum=0.9),
      loss=tf.keras.losses.CategoricalCrossentropy(),
      metrics=['accuracy']
    )
```

```
[ ]: history = model.fit(
      x=train_ds,
      batch_size=64,
      epochs=150,
      verbose=1,
      validation_data=test_ds
    )
```

Epoch 1/150

```

2022-11-04 00:23:50.661299: I
tensorflow/core/grappler/optimizers/custom_graph_optimizer_registry.cc:114]
Plugin optimizer for device_type GPU is enabled.

12/12 [=====] - 1s 49ms/step - loss: 1.3532 - accuracy:
0.3197 - val_loss: 1.2619 - val_accuracy: 0.3386

2022-11-04 00:23:51.307950: I

```

tensorflow/core/grappler/optimizers/custom_graph_optimizer_registry.cc:114]

Plugin optimizer for device_type GPU is enabled.

Epoch 2/150

12/12 [=====] - 0s 23ms/step - loss: 1.3014 - accuracy: 0.3184 - val_loss: 1.1909 - val_accuracy: 0.3545

Epoch 3/150

12/12 [=====] - 0s 23ms/step - loss: 1.2639 - accuracy: 0.3382 - val_loss: 1.1372 - val_accuracy: 0.3598

Epoch 4/150

12/12 [=====] - 0s 24ms/step - loss: 1.1526 - accuracy: 0.3765 - val_loss: 1.1013 - val_accuracy: 0.3757

Epoch 5/150

12/12 [=====] - 0s 23ms/step - loss: 1.1809 - accuracy: 0.3712 - val_loss: 1.0741 - val_accuracy: 0.3810

Epoch 6/150

12/12 [=====] - 0s 20ms/step - loss: 1.0958 - accuracy: 0.4069 - val_loss: 1.0501 - val_accuracy: 0.4286

Epoch 7/150

12/12 [=====] - 0s 20ms/step - loss: 1.0892 - accuracy: 0.4597 - val_loss: 1.0317 - val_accuracy: 0.4444

Epoch 8/150

12/12 [=====] - 0s 24ms/step - loss: 1.0416 - accuracy: 0.4597 - val_loss: 1.0135 - val_accuracy: 0.4656

Epoch 9/150

12/12 [=====] - 0s 21ms/step - loss: 1.0308 - accuracy: 0.4624 - val_loss: 1.0008 - val_accuracy: 0.4656

Epoch 10/150

12/12 [=====] - 0s 24ms/step - loss: 1.0228 - accuracy: 0.4795 - val_loss: 0.9898 - val_accuracy: 0.4974

Epoch 11/150

12/12 [=====] - 0s 22ms/step - loss: 0.9863 - accuracy: 0.5099 - val_loss: 0.9807 - val_accuracy: 0.5185

Epoch 12/150

12/12 [=====] - 0s 22ms/step - loss: 0.9804 - accuracy: 0.5244 - val_loss: 0.9718 - val_accuracy: 0.5450

Epoch 13/150

12/12 [=====] - 0s 21ms/step - loss: 0.9783 - accuracy: 0.5429 - val_loss: 0.9660 - val_accuracy: 0.5344

Epoch 14/150

12/12 [=====] - 0s 20ms/step - loss: 0.9654 - accuracy: 0.5139 - val_loss: 0.9609 - val_accuracy: 0.5397

Epoch 15/150

12/12 [=====] - 0s 20ms/step - loss: 0.9719 - accuracy: 0.5192 - val_loss: 0.9558 - val_accuracy: 0.5450

Epoch 16/150

12/12 [=====] - 0s 20ms/step - loss: 0.9297 - accuracy: 0.5522 - val_loss: 0.9507 - val_accuracy: 0.5450

Epoch 17/150
12/12 [=====] - 0s 20ms/step - loss: 0.9494 - accuracy:
0.5376 - val_loss: 0.9460 - val_accuracy: 0.5503
Epoch 18/150
12/12 [=====] - 0s 21ms/step - loss: 0.9391 - accuracy:
0.5297 - val_loss: 0.9432 - val_accuracy: 0.5503
Epoch 19/150
12/12 [=====] - 0s 20ms/step - loss: 0.9291 - accuracy:
0.5495 - val_loss: 0.9411 - val_accuracy: 0.5503
Epoch 20/150
12/12 [=====] - 0s 21ms/step - loss: 0.9340 - accuracy:
0.5627 - val_loss: 0.9398 - val_accuracy: 0.5397
Epoch 21/150
12/12 [=====] - 0s 21ms/step - loss: 0.9274 - accuracy:
0.5812 - val_loss: 0.9379 - val_accuracy: 0.5397
Epoch 22/150
12/12 [=====] - 0s 21ms/step - loss: 0.9071 - accuracy:
0.5575 - val_loss: 0.9365 - val_accuracy: 0.5344
Epoch 23/150
12/12 [=====] - 0s 22ms/step - loss: 0.8915 - accuracy:
0.5945 - val_loss: 0.9332 - val_accuracy: 0.5450
Epoch 24/150
12/12 [=====] - 0s 23ms/step - loss: 0.8856 - accuracy:
0.5931 - val_loss: 0.9297 - val_accuracy: 0.5450
Epoch 25/150
12/12 [=====] - 0s 21ms/step - loss: 0.9158 - accuracy:
0.5627 - val_loss: 0.9282 - val_accuracy: 0.5450
Epoch 26/150
12/12 [=====] - 0s 22ms/step - loss: 0.8609 - accuracy:
0.5865 - val_loss: 0.9260 - val_accuracy: 0.5503
Epoch 27/150
12/12 [=====] - 0s 20ms/step - loss: 0.8716 - accuracy:
0.5865 - val_loss: 0.9252 - val_accuracy: 0.5503
Epoch 28/150
12/12 [=====] - 0s 20ms/step - loss: 0.8784 - accuracy:
0.5918 - val_loss: 0.9230 - val_accuracy: 0.5397
Epoch 29/150
12/12 [=====] - 0s 20ms/step - loss: 0.8907 - accuracy:
0.5878 - val_loss: 0.9210 - val_accuracy: 0.5450
Epoch 30/150
12/12 [=====] - 0s 20ms/step - loss: 0.8975 - accuracy:
0.5720 - val_loss: 0.9188 - val_accuracy: 0.5608
Epoch 31/150
12/12 [=====] - 0s 20ms/step - loss: 0.8849 - accuracy:
0.5918 - val_loss: 0.9168 - val_accuracy: 0.5714
Epoch 32/150
12/12 [=====] - 0s 21ms/step - loss: 0.8529 - accuracy:
0.5931 - val_loss: 0.9151 - val_accuracy: 0.5767

Epoch 33/150
12/12 [=====] - 0s 20ms/step - loss: 0.8573 - accuracy:
0.6182 - val_loss: 0.9144 - val_accuracy: 0.5714
Epoch 34/150
12/12 [=====] - 0s 22ms/step - loss: 0.8529 - accuracy:
0.5958 - val_loss: 0.9130 - val_accuracy: 0.5714
Epoch 35/150
12/12 [=====] - 0s 23ms/step - loss: 0.8494 - accuracy:
0.6024 - val_loss: 0.9140 - val_accuracy: 0.5608
Epoch 36/150
12/12 [=====] - 0s 21ms/step - loss: 0.8639 - accuracy:
0.5958 - val_loss: 0.9120 - val_accuracy: 0.5556
Epoch 37/150
12/12 [=====] - 0s 20ms/step - loss: 0.8528 - accuracy:
0.6011 - val_loss: 0.9092 - val_accuracy: 0.5661
Epoch 38/150
12/12 [=====] - 0s 24ms/step - loss: 0.8607 - accuracy:
0.6116 - val_loss: 0.9073 - val_accuracy: 0.5714
Epoch 39/150
12/12 [=====] - 0s 22ms/step - loss: 0.8349 - accuracy:
0.6156 - val_loss: 0.9058 - val_accuracy: 0.5820
Epoch 40/150
12/12 [=====] - 0s 22ms/step - loss: 0.8552 - accuracy:
0.6341 - val_loss: 0.9072 - val_accuracy: 0.5714
Epoch 41/150
12/12 [=====] - 0s 21ms/step - loss: 0.8588 - accuracy:
0.6143 - val_loss: 0.9058 - val_accuracy: 0.5820
Epoch 42/150
12/12 [=====] - 0s 19ms/step - loss: 0.8664 - accuracy:
0.5971 - val_loss: 0.9042 - val_accuracy: 0.5767
Epoch 43/150
12/12 [=====] - 0s 20ms/step - loss: 0.8424 - accuracy:
0.6222 - val_loss: 0.9015 - val_accuracy: 0.5767
Epoch 44/150
12/12 [=====] - 0s 20ms/step - loss: 0.8321 - accuracy:
0.6288 - val_loss: 0.8995 - val_accuracy: 0.5820
Epoch 45/150
12/12 [=====] - 0s 20ms/step - loss: 0.8544 - accuracy:
0.6156 - val_loss: 0.8986 - val_accuracy: 0.5820
Epoch 46/150
12/12 [=====] - 0s 20ms/step - loss: 0.8517 - accuracy:
0.5984 - val_loss: 0.8982 - val_accuracy: 0.5873
Epoch 47/150
12/12 [=====] - 0s 20ms/step - loss: 0.8344 - accuracy:
0.6209 - val_loss: 0.8978 - val_accuracy: 0.5873
Epoch 48/150
12/12 [=====] - 0s 20ms/step - loss: 0.8565 - accuracy:
0.5997 - val_loss: 0.8955 - val_accuracy: 0.5767

Epoch 49/150
12/12 [=====] - 0s 21ms/step - loss: 0.8101 - accuracy:
0.6275 - val_loss: 0.8951 - val_accuracy: 0.5820
Epoch 50/150
12/12 [=====] - 0s 20ms/step - loss: 0.8410 - accuracy:
0.6248 - val_loss: 0.8932 - val_accuracy: 0.5873
Epoch 51/150
12/12 [=====] - 0s 21ms/step - loss: 0.8146 - accuracy:
0.6275 - val_loss: 0.8930 - val_accuracy: 0.5873
Epoch 52/150
12/12 [=====] - 0s 22ms/step - loss: 0.8382 - accuracy:
0.6050 - val_loss: 0.8932 - val_accuracy: 0.5873
Epoch 53/150
12/12 [=====] - 0s 21ms/step - loss: 0.8204 - accuracy:
0.6182 - val_loss: 0.8939 - val_accuracy: 0.5873
Epoch 54/150
12/12 [=====] - 0s 20ms/step - loss: 0.8302 - accuracy:
0.6116 - val_loss: 0.8927 - val_accuracy: 0.5873
Epoch 55/150
12/12 [=====] - 0s 20ms/step - loss: 0.7954 - accuracy:
0.6592 - val_loss: 0.8908 - val_accuracy: 0.5926
Epoch 56/150
12/12 [=====] - 0s 21ms/step - loss: 0.8248 - accuracy:
0.6288 - val_loss: 0.8891 - val_accuracy: 0.5926
Epoch 57/150
12/12 [=====] - 0s 21ms/step - loss: 0.8172 - accuracy:
0.6367 - val_loss: 0.8878 - val_accuracy: 0.6032
Epoch 58/150
12/12 [=====] - 0s 20ms/step - loss: 0.8277 - accuracy:
0.6433 - val_loss: 0.8876 - val_accuracy: 0.5926
Epoch 59/150
12/12 [=====] - 0s 21ms/step - loss: 0.8146 - accuracy:
0.6314 - val_loss: 0.8863 - val_accuracy: 0.5926
Epoch 60/150
12/12 [=====] - 0s 21ms/step - loss: 0.8021 - accuracy:
0.6209 - val_loss: 0.8860 - val_accuracy: 0.5926
Epoch 61/150
12/12 [=====] - 0s 21ms/step - loss: 0.8161 - accuracy:
0.6328 - val_loss: 0.8849 - val_accuracy: 0.5926
Epoch 62/150
12/12 [=====] - 0s 20ms/step - loss: 0.8067 - accuracy:
0.6486 - val_loss: 0.8845 - val_accuracy: 0.5926
Epoch 63/150
12/12 [=====] - 0s 20ms/step - loss: 0.8361 - accuracy:
0.6275 - val_loss: 0.8846 - val_accuracy: 0.5926
Epoch 64/150
12/12 [=====] - 0s 19ms/step - loss: 0.7995 - accuracy:
0.6433 - val_loss: 0.8845 - val_accuracy: 0.5926

Epoch 65/150
12/12 [=====] - 0s 20ms/step - loss: 0.7912 - accuracy:
0.6526 - val_loss: 0.8839 - val_accuracy: 0.5926
Epoch 66/150
12/12 [=====] - 0s 20ms/step - loss: 0.8219 - accuracy:
0.6446 - val_loss: 0.8829 - val_accuracy: 0.5926
Epoch 67/150
12/12 [=====] - 0s 20ms/step - loss: 0.8012 - accuracy:
0.6433 - val_loss: 0.8839 - val_accuracy: 0.5926
Epoch 68/150
12/12 [=====] - 0s 20ms/step - loss: 0.8251 - accuracy:
0.6314 - val_loss: 0.8824 - val_accuracy: 0.5926
Epoch 69/150
12/12 [=====] - 0s 21ms/step - loss: 0.8249 - accuracy:
0.6103 - val_loss: 0.8815 - val_accuracy: 0.5873
Epoch 70/150
12/12 [=====] - 0s 21ms/step - loss: 0.7926 - accuracy:
0.6631 - val_loss: 0.8809 - val_accuracy: 0.6032
Epoch 71/150
12/12 [=====] - 0s 21ms/step - loss: 0.8066 - accuracy:
0.6341 - val_loss: 0.8789 - val_accuracy: 0.6085
Epoch 72/150
12/12 [=====] - 0s 21ms/step - loss: 0.8205 - accuracy:
0.6341 - val_loss: 0.8781 - val_accuracy: 0.6032
Epoch 73/150
12/12 [=====] - 0s 20ms/step - loss: 0.8130 - accuracy:
0.6380 - val_loss: 0.8761 - val_accuracy: 0.6085
Epoch 74/150
12/12 [=====] - 0s 20ms/step - loss: 0.7952 - accuracy:
0.6473 - val_loss: 0.8769 - val_accuracy: 0.6085
Epoch 75/150
12/12 [=====] - 0s 20ms/step - loss: 0.8146 - accuracy:
0.6407 - val_loss: 0.8779 - val_accuracy: 0.6138
Epoch 76/150
12/12 [=====] - 0s 21ms/step - loss: 0.7770 - accuracy:
0.6711 - val_loss: 0.8765 - val_accuracy: 0.6085
Epoch 77/150
12/12 [=====] - 0s 20ms/step - loss: 0.8175 - accuracy:
0.6262 - val_loss: 0.8748 - val_accuracy: 0.6085
Epoch 78/150
12/12 [=====] - 0s 20ms/step - loss: 0.7919 - accuracy:
0.6737 - val_loss: 0.8757 - val_accuracy: 0.5873
Epoch 79/150
12/12 [=====] - 0s 20ms/step - loss: 0.8196 - accuracy:
0.6367 - val_loss: 0.8747 - val_accuracy: 0.6243
Epoch 80/150
12/12 [=====] - 0s 22ms/step - loss: 0.7903 - accuracy:
0.6711 - val_loss: 0.8737 - val_accuracy: 0.6138

Epoch 81/150
12/12 [=====] - 0s 22ms/step - loss: 0.8179 - accuracy:
0.6248 - val_loss: 0.8729 - val_accuracy: 0.6138
Epoch 82/150
12/12 [=====] - 0s 23ms/step - loss: 0.8164 - accuracy:
0.6182 - val_loss: 0.8733 - val_accuracy: 0.6243
Epoch 83/150
12/12 [=====] - 0s 21ms/step - loss: 0.8170 - accuracy:
0.6446 - val_loss: 0.8734 - val_accuracy: 0.6138
Epoch 84/150
12/12 [=====] - 0s 21ms/step - loss: 0.8035 - accuracy:
0.6407 - val_loss: 0.8722 - val_accuracy: 0.6138
Epoch 85/150
12/12 [=====] - 0s 21ms/step - loss: 0.8070 - accuracy:
0.6407 - val_loss: 0.8713 - val_accuracy: 0.6190
Epoch 86/150
12/12 [=====] - 0s 21ms/step - loss: 0.8062 - accuracy:
0.6354 - val_loss: 0.8714 - val_accuracy: 0.6138
Epoch 87/150
12/12 [=====] - 0s 21ms/step - loss: 0.8127 - accuracy:
0.6394 - val_loss: 0.8708 - val_accuracy: 0.6138
Epoch 88/150
12/12 [=====] - 0s 20ms/step - loss: 0.7921 - accuracy:
0.6486 - val_loss: 0.8686 - val_accuracy: 0.6243
Epoch 89/150
12/12 [=====] - 0s 20ms/step - loss: 0.8170 - accuracy:
0.6354 - val_loss: 0.8679 - val_accuracy: 0.6085
Epoch 90/150
12/12 [=====] - 0s 20ms/step - loss: 0.7895 - accuracy:
0.6407 - val_loss: 0.8675 - val_accuracy: 0.6085
Epoch 91/150
12/12 [=====] - 0s 20ms/step - loss: 0.7687 - accuracy:
0.6764 - val_loss: 0.8672 - val_accuracy: 0.6138
Epoch 92/150
12/12 [=====] - 0s 20ms/step - loss: 0.7802 - accuracy:
0.6486 - val_loss: 0.8659 - val_accuracy: 0.6085
Epoch 93/150
12/12 [=====] - 0s 19ms/step - loss: 0.7906 - accuracy:
0.6460 - val_loss: 0.8667 - val_accuracy: 0.6085
Epoch 94/150
12/12 [=====] - 0s 21ms/step - loss: 0.8157 - accuracy:
0.6341 - val_loss: 0.8651 - val_accuracy: 0.6243
Epoch 95/150
12/12 [=====] - 0s 20ms/step - loss: 0.7735 - accuracy:
0.6513 - val_loss: 0.8637 - val_accuracy: 0.6243
Epoch 96/150
12/12 [=====] - 0s 20ms/step - loss: 0.7790 - accuracy:
0.6645 - val_loss: 0.8632 - val_accuracy: 0.6190

Epoch 97/150
12/12 [=====] - 0s 20ms/step - loss: 0.7816 - accuracy:
0.6486 - val_loss: 0.8631 - val_accuracy: 0.6138
Epoch 98/150
12/12 [=====] - 0s 21ms/step - loss: 0.7943 - accuracy:
0.6605 - val_loss: 0.8620 - val_accuracy: 0.6138
Epoch 99/150
12/12 [=====] - 0s 23ms/step - loss: 0.8071 - accuracy:
0.6592 - val_loss: 0.8609 - val_accuracy: 0.6138
Epoch 100/150
12/12 [=====] - 0s 22ms/step - loss: 0.7923 - accuracy:
0.6486 - val_loss: 0.8616 - val_accuracy: 0.6138
Epoch 101/150
12/12 [=====] - 0s 22ms/step - loss: 0.7832 - accuracy:
0.6618 - val_loss: 0.8624 - val_accuracy: 0.6032
Epoch 102/150
12/12 [=====] - 0s 21ms/step - loss: 0.7801 - accuracy:
0.6539 - val_loss: 0.8620 - val_accuracy: 0.6085
Epoch 103/150
12/12 [=====] - 0s 21ms/step - loss: 0.7847 - accuracy:
0.6711 - val_loss: 0.8606 - val_accuracy: 0.6190
Epoch 104/150
12/12 [=====] - 0s 21ms/step - loss: 0.7819 - accuracy:
0.6579 - val_loss: 0.8600 - val_accuracy: 0.6138
Epoch 105/150
12/12 [=====] - 0s 20ms/step - loss: 0.7987 - accuracy:
0.6499 - val_loss: 0.8593 - val_accuracy: 0.6138
Epoch 106/150
12/12 [=====] - 0s 20ms/step - loss: 0.7890 - accuracy:
0.6565 - val_loss: 0.8582 - val_accuracy: 0.6190
Epoch 107/150
12/12 [=====] - 0s 21ms/step - loss: 0.7783 - accuracy:
0.6460 - val_loss: 0.8577 - val_accuracy: 0.6243
Epoch 108/150
12/12 [=====] - 0s 22ms/step - loss: 0.7829 - accuracy:
0.6486 - val_loss: 0.8579 - val_accuracy: 0.6138
Epoch 109/150
12/12 [=====] - 0s 22ms/step - loss: 0.7762 - accuracy:
0.6830 - val_loss: 0.8570 - val_accuracy: 0.6190
Epoch 110/150
12/12 [=====] - 0s 23ms/step - loss: 0.7976 - accuracy:
0.6460 - val_loss: 0.8572 - val_accuracy: 0.6190
Epoch 111/150
12/12 [=====] - 0s 22ms/step - loss: 0.8048 - accuracy:
0.6380 - val_loss: 0.8564 - val_accuracy: 0.6190
Epoch 112/150
12/12 [=====] - 0s 22ms/step - loss: 0.7697 - accuracy:
0.6764 - val_loss: 0.8553 - val_accuracy: 0.6190

Epoch 113/150
12/12 [=====] - 0s 21ms/step - loss: 0.7855 - accuracy: 0.6697 - val_loss: 0.8553 - val_accuracy: 0.6138

Epoch 114/150
12/12 [=====] - 0s 21ms/step - loss: 0.7605 - accuracy: 0.6750 - val_loss: 0.8538 - val_accuracy: 0.6243

Epoch 115/150
12/12 [=====] - 0s 23ms/step - loss: 0.7830 - accuracy: 0.6605 - val_loss: 0.8535 - val_accuracy: 0.6190

Epoch 116/150
12/12 [=====] - 0s 22ms/step - loss: 0.7617 - accuracy: 0.6526 - val_loss: 0.8539 - val_accuracy: 0.6138

Epoch 117/150
12/12 [=====] - 0s 21ms/step - loss: 0.7753 - accuracy: 0.6565 - val_loss: 0.8542 - val_accuracy: 0.6085

Epoch 118/150
12/12 [=====] - 0s 24ms/step - loss: 0.7631 - accuracy: 0.6697 - val_loss: 0.8526 - val_accuracy: 0.6138

Epoch 119/150
12/12 [=====] - 0s 22ms/step - loss: 0.7725 - accuracy: 0.6460 - val_loss: 0.8520 - val_accuracy: 0.6190

Epoch 120/150
12/12 [=====] - 0s 21ms/step - loss: 0.7624 - accuracy: 0.6552 - val_loss: 0.8513 - val_accuracy: 0.6190

Epoch 121/150
12/12 [=====] - 0s 22ms/step - loss: 0.7895 - accuracy: 0.6433 - val_loss: 0.8507 - val_accuracy: 0.6190

Epoch 122/150
12/12 [=====] - 0s 22ms/step - loss: 0.7621 - accuracy: 0.6605 - val_loss: 0.8524 - val_accuracy: 0.6085

Epoch 123/150
12/12 [=====] - 0s 21ms/step - loss: 0.7630 - accuracy: 0.6631 - val_loss: 0.8515 - val_accuracy: 0.6085

Epoch 124/150
12/12 [=====] - 0s 21ms/step - loss: 0.7675 - accuracy: 0.6579 - val_loss: 0.8498 - val_accuracy: 0.6085

Epoch 125/150
12/12 [=====] - 0s 21ms/step - loss: 0.7498 - accuracy: 0.6671 - val_loss: 0.8482 - val_accuracy: 0.6190

Epoch 126/150
12/12 [=====] - 0s 20ms/step - loss: 0.8034 - accuracy: 0.6446 - val_loss: 0.8472 - val_accuracy: 0.6190

Epoch 127/150
12/12 [=====] - 0s 21ms/step - loss: 0.8040 - accuracy: 0.6526 - val_loss: 0.8466 - val_accuracy: 0.6243

Epoch 128/150
12/12 [=====] - 0s 20ms/step - loss: 0.7667 - accuracy: 0.6671 - val_loss: 0.8465 - val_accuracy: 0.6243

Epoch 129/150
12/12 [=====] - 0s 20ms/step - loss: 0.7619 - accuracy:
0.6684 - val_loss: 0.8471 - val_accuracy: 0.6190

Epoch 130/150
12/12 [=====] - 0s 22ms/step - loss: 0.7732 - accuracy:
0.6645 - val_loss: 0.8467 - val_accuracy: 0.6243

Epoch 131/150
12/12 [=====] - 0s 22ms/step - loss: 0.7782 - accuracy:
0.6446 - val_loss: 0.8456 - val_accuracy: 0.6243

Epoch 132/150
12/12 [=====] - 0s 21ms/step - loss: 0.7597 - accuracy:
0.6882 - val_loss: 0.8451 - val_accuracy: 0.6243

Epoch 133/150
12/12 [=====] - 0s 22ms/step - loss: 0.7874 - accuracy:
0.6552 - val_loss: 0.8446 - val_accuracy: 0.6190

Epoch 134/150
12/12 [=====] - 0s 21ms/step - loss: 0.7887 - accuracy:
0.6526 - val_loss: 0.8449 - val_accuracy: 0.6190

Epoch 135/150
12/12 [=====] - 0s 21ms/step - loss: 0.7838 - accuracy:
0.6513 - val_loss: 0.8451 - val_accuracy: 0.6243

Epoch 136/150
12/12 [=====] - 0s 22ms/step - loss: 0.7464 - accuracy:
0.6737 - val_loss: 0.8454 - val_accuracy: 0.6190

Epoch 137/150
12/12 [=====] - 0s 21ms/step - loss: 0.7375 - accuracy:
0.6816 - val_loss: 0.8449 - val_accuracy: 0.6243

Epoch 138/150
12/12 [=====] - 0s 21ms/step - loss: 0.7763 - accuracy:
0.6328 - val_loss: 0.8438 - val_accuracy: 0.6190

Epoch 139/150
12/12 [=====] - 0s 21ms/step - loss: 0.7519 - accuracy:
0.6869 - val_loss: 0.8432 - val_accuracy: 0.6243

Epoch 140/150
12/12 [=====] - 0s 22ms/step - loss: 0.7708 - accuracy:
0.6605 - val_loss: 0.8434 - val_accuracy: 0.6138

Epoch 141/150
12/12 [=====] - 0s 23ms/step - loss: 0.7483 - accuracy:
0.6830 - val_loss: 0.8442 - val_accuracy: 0.6190

Epoch 142/150
12/12 [=====] - 0s 20ms/step - loss: 0.7574 - accuracy:
0.6764 - val_loss: 0.8433 - val_accuracy: 0.6243

Epoch 143/150
12/12 [=====] - 0s 22ms/step - loss: 0.7597 - accuracy:
0.6618 - val_loss: 0.8423 - val_accuracy: 0.6243

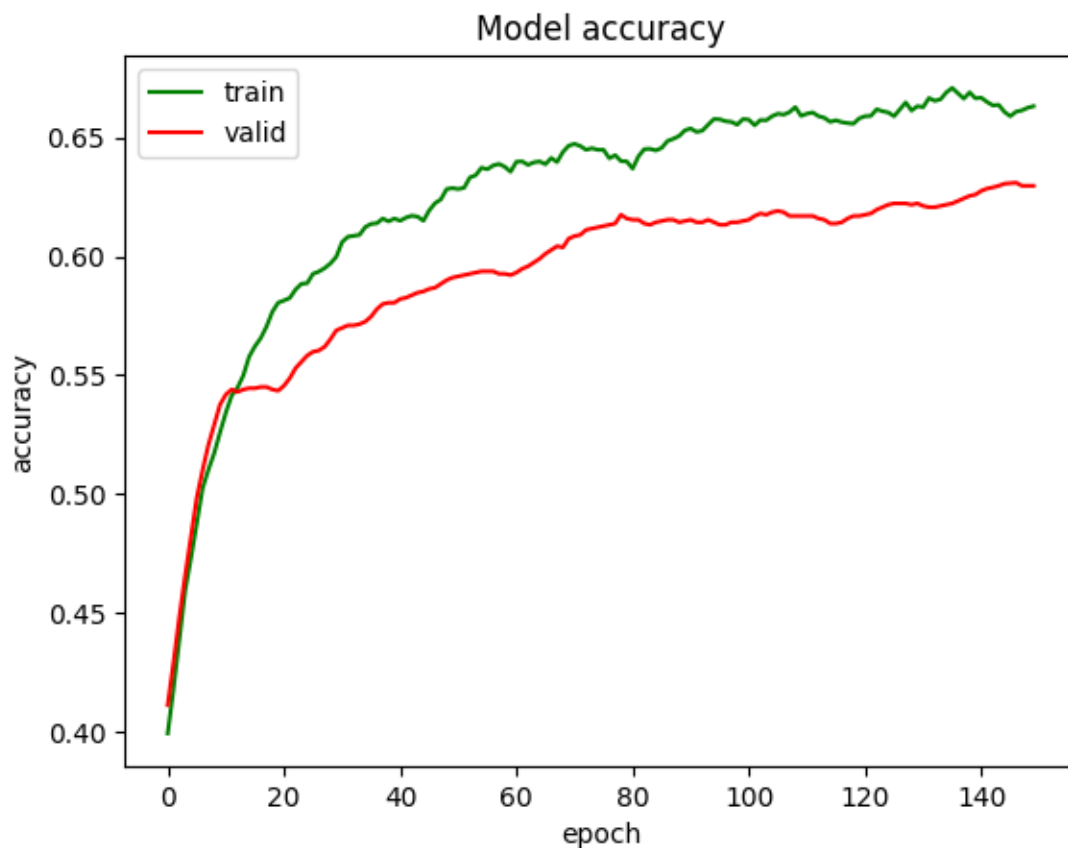
Epoch 144/150
12/12 [=====] - 0s 21ms/step - loss: 0.7542 - accuracy:
0.6816 - val_loss: 0.8417 - val_accuracy: 0.6243

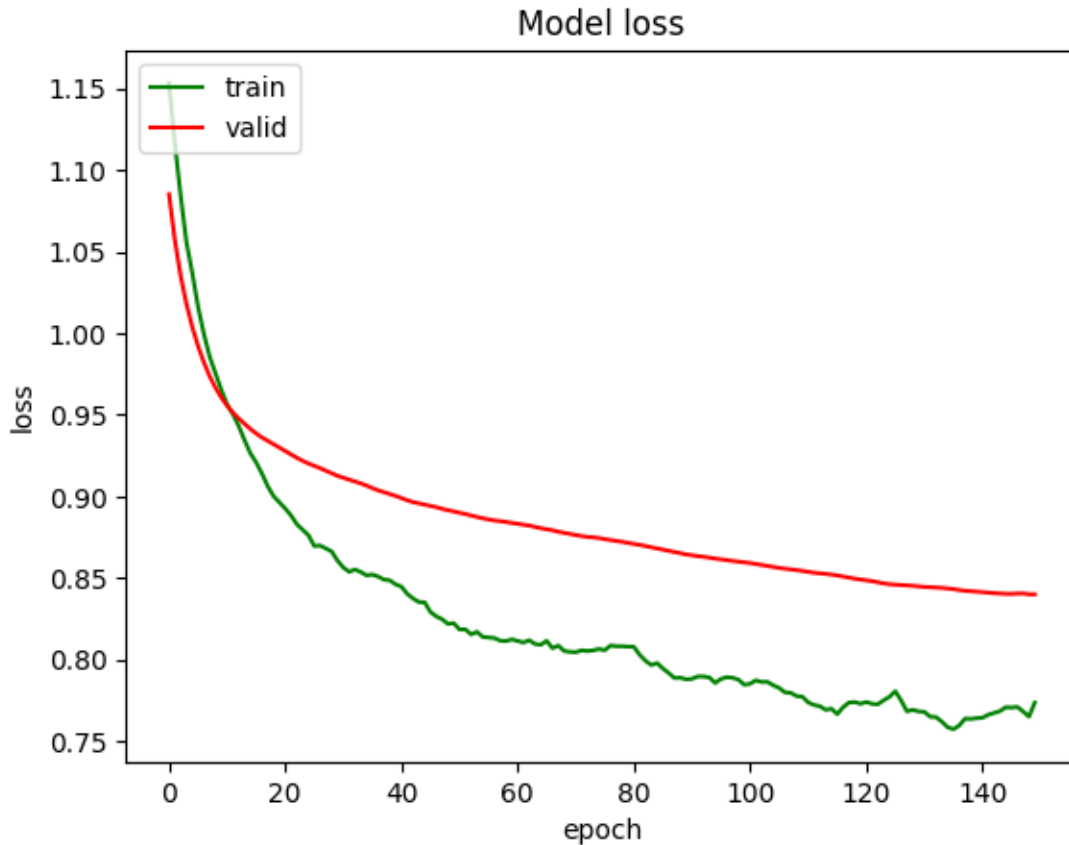
```

Epoch 145/150
12/12 [=====] - 0s 22ms/step - loss: 0.7717 - accuracy:
0.6697 - val_loss: 0.8407 - val_accuracy: 0.6296
Epoch 146/150
12/12 [=====] - 0s 21ms/step - loss: 0.7692 - accuracy:
0.6513 - val_loss: 0.8394 - val_accuracy: 0.6296
Epoch 147/150
12/12 [=====] - 0s 24ms/step - loss: 0.7796 - accuracy:
0.6592 - val_loss: 0.8404 - val_accuracy: 0.6349
Epoch 148/150
12/12 [=====] - 0s 22ms/step - loss: 0.7746 - accuracy:
0.6592 - val_loss: 0.8415 - val_accuracy: 0.6296
Epoch 149/150
12/12 [=====] - 0s 21ms/step - loss: 0.7566 - accuracy:
0.6618 - val_loss: 0.8402 - val_accuracy: 0.6296
Epoch 150/150
12/12 [=====] - 0s 22ms/step - loss: 0.7737 - accuracy:
0.6631 - val_loss: 0.8401 - val_accuracy: 0.6296

```

```
[ ]: print_history(history.history)
```





```
[ ]: for ix in range(len(model.layers[:-1])):
      assert model.layers[ix].weights == old_weights[ix]
```

Jak widać model zachowuje się tak jak powinien - tylko ostatnia warstwa zmienia wagi. Niestety wyniki są poniżej oczekiwań, szczególnie w porównaniu do trenowania modelu od zera. Możliwe, że mógłbym dobrać lepsze parametry, np. learning rate, liczbę epok, czy momentum.

3 ImageNet

```
[ ]: train_ds, test_ds, _ = load_ds(image_size=(256,256))
```

Found 946 files belonging to 3 classes.
Using 757 files for training.
Using 189 files for validation.

```
[ ]: model = tf.keras.applications.xception.Xception(
      include_top=False,
      input_shape=(256,256,3),
      # pooling='avg',
      classifier_activation='softmax')
```

```
)
```

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

```
Model: "xception"
```

```
-----
Layer (type)                 Output Shape              Param #   Connected to
=====
input_1 (InputLayer)         [(None, 256, 256, 3      0      []
                                )]

block1_conv1 (Conv2D)         (None, 127, 127, 32      864
['input_1[0][0]'])

block1_conv1_bn (BatchNormaliz (None, 127, 127, 32      128
['block1_conv1[0][0]']
ation)

block1_conv1_act (Activation) (None, 127, 127, 32      0
['block1_conv1_bn[0][0]'])

block1_conv2 (Conv2D)         (None, 125, 125, 64      18432
['block1_conv1_act[0][0]'])

block1_conv2_bn (BatchNormaliz (None, 125, 125, 64      256
['block1_conv2[0][0]']
ation)

block1_conv2_act (Activation) (None, 125, 125, 64      0
['block1_conv2_bn[0][0]'])

block2_sepconv1 (SeparableConv (None, 125, 125, 12      8768
['block1_conv2_act[0][0]']
2D)

block2_sepconv1_bn (BatchNorma (None, 125, 125, 12      512
['block2_sepconv1[0][0]']
lization)

block2_sepconv2_act (Activatio (None, 125, 125, 12      0
['block2_sepconv1_bn[0][0]']
n)
```

```

block2_sepconv2 (SeparableConv (None, 125, 125, 12 17536
['block2_sepconv2_act[0][0]']
2D) 8)

block2_sepconv2_bn (BatchNorma (None, 125, 125, 12 512
['block2_sepconv2[0][0]']
lization) 8)

conv2d_16 (Conv2D) (None, 63, 63, 128) 8192
['block1_conv2_act[0][0]']

block2_pool (MaxPooling2D) (None, 63, 63, 128) 0
['block2_sepconv2_bn[0][0]']

batch_normalization_16 (BatchN (None, 63, 63, 128) 512
['conv2d_16[0][0]']
ormalization)

add (Add) (None, 63, 63, 128) 0
['block2_pool[0][0]',
'batch_normalization_16[0][0]']

block3_sepconv1_act (Activatio (None, 63, 63, 128) 0 ['add[0][0]']
n)

block3_sepconv1 (SeparableConv (None, 63, 63, 256) 33920
['block3_sepconv1_act[0][0]']
2D)

block3_sepconv1_bn (BatchNorma (None, 63, 63, 256) 1024
['block3_sepconv1[0][0]']
lization)

block3_sepconv2_act (Activatio (None, 63, 63, 256) 0
['block3_sepconv1_bn[0][0]']
n)

block3_sepconv2 (SeparableConv (None, 63, 63, 256) 67840
['block3_sepconv2_act[0][0]']
2D)

block3_sepconv2_bn (BatchNorma (None, 63, 63, 256) 1024
['block3_sepconv2[0][0]']
lization)

conv2d_17 (Conv2D) (None, 32, 32, 256) 32768 ['add[0][0]']

```

```

block3_pool (MaxPooling2D)      (None, 32, 32, 256)  0
['block3_sepconv2_bn[0][0]']

batch_normalization_17 (BatchN  (None, 32, 32, 256)  1024
['conv2d_17[0][0]']
ormalization)

add_1 (Add)                      (None, 32, 32, 256)  0
['block3_pool[0][0]',
'batch_normalization_17[0][0]']

block4_sepconv1_act (Activatio  (None, 32, 32, 256)  0          ['add_1[0][0]']
n)

block4_sepconv1 (SeparableConv  (None, 32, 32, 728)  188672
['block4_sepconv1_act[0][0]']
2D)

block4_sepconv1_bn (BatchNorma  (None, 32, 32, 728)  2912
['block4_sepconv1[0][0]']
lization)

block4_sepconv2_act (Activatio  (None, 32, 32, 728)  0
['block4_sepconv1_bn[0][0]']
n)

block4_sepconv2 (SeparableConv  (None, 32, 32, 728)  536536
['block4_sepconv2_act[0][0]']
2D)

block4_sepconv2_bn (BatchNorma  (None, 32, 32, 728)  2912
['block4_sepconv2[0][0]']
lization)

conv2d_18 (Conv2D)              (None, 16, 16, 728)  186368          ['add_1[0][0]']

block4_pool (MaxPooling2D)      (None, 16, 16, 728)  0
['block4_sepconv2_bn[0][0]']

batch_normalization_18 (BatchN  (None, 16, 16, 728)  2912
['conv2d_18[0][0]']
ormalization)

add_2 (Add)                      (None, 16, 16, 728)  0
['block4_pool[0][0]',
'batch_normalization_18[0][0]']

block5_sepconv1_act (Activatio  (None, 16, 16, 728)  0          ['add_2[0][0]']

```

```

n)

block5_sepconv1 (SeparableConv (None, 16, 16, 728) 536536
['block5_sepconv1_act[0][0]']
2D)

block5_sepconv1_bn (BatchNorma (None, 16, 16, 728) 2912
['block5_sepconv1[0][0]']
lization)

block5_sepconv2_act (Activatio (None, 16, 16, 728) 0
['block5_sepconv1_bn[0][0]']
n)

block5_sepconv2 (SeparableConv (None, 16, 16, 728) 536536
['block5_sepconv2_act[0][0]']
2D)

block5_sepconv2_bn (BatchNorma (None, 16, 16, 728) 2912
['block5_sepconv2[0][0]']
lization)

block5_sepconv3_act (Activatio (None, 16, 16, 728) 0
['block5_sepconv2_bn[0][0]']
n)

block5_sepconv3 (SeparableConv (None, 16, 16, 728) 536536
['block5_sepconv3_act[0][0]']
2D)

block5_sepconv3_bn (BatchNorma (None, 16, 16, 728) 2912
['block5_sepconv3[0][0]']
lization)

add_3 (Add) (None, 16, 16, 728) 0
['block5_sepconv3_bn[0][0]',
'add_2[0][0]']

block6_sepconv1_act (Activatio (None, 16, 16, 728) 0 ['add_3[0][0]']
n)

block6_sepconv1 (SeparableConv (None, 16, 16, 728) 536536
['block6_sepconv1_act[0][0]']
2D)

block6_sepconv1_bn (BatchNorma (None, 16, 16, 728) 2912
['block6_sepconv1[0][0]']
lization)

```

```

block6_sepconv2_act (Activation (None, 16, 16, 728) 0
['block6_sepconv1_bn[0][0]']
n)

block6_sepconv2 (SeparableConv (None, 16, 16, 728) 536536
['block6_sepconv2_act[0][0]']
2D)

block6_sepconv2_bn (BatchNormaliza (None, 16, 16, 728) 2912
['block6_sepconv2[0][0]']
lization)

block6_sepconv3_act (Activation (None, 16, 16, 728) 0
['block6_sepconv2_bn[0][0]']
n)

block6_sepconv3 (SeparableConv (None, 16, 16, 728) 536536
['block6_sepconv3_act[0][0]']
2D)

block6_sepconv3_bn (BatchNormaliza (None, 16, 16, 728) 2912
['block6_sepconv3[0][0]']
lization)

add_4 (Add) (None, 16, 16, 728) 0
['block6_sepconv3_bn[0][0]',
'add_3[0][0]']

block7_sepconv1_act (Activation (None, 16, 16, 728) 0 ['add_4[0][0]']
n)

block7_sepconv1 (SeparableConv (None, 16, 16, 728) 536536
['block7_sepconv1_act[0][0]']
2D)

block7_sepconv1_bn (BatchNormaliza (None, 16, 16, 728) 2912
['block7_sepconv1[0][0]']
lization)

block7_sepconv2_act (Activation (None, 16, 16, 728) 0
['block7_sepconv1_bn[0][0]']
n)

block7_sepconv2 (SeparableConv (None, 16, 16, 728) 536536
['block7_sepconv2_act[0][0]']
2D)

```

```

block7_sepconv2_bn (BatchNorma (None, 16, 16, 728) 2912
['block7_sepconv2[0][0]']
lization)

block7_sepconv3_act (Activatio (None, 16, 16, 728) 0
['block7_sepconv2_bn[0][0]']
n)

block7_sepconv3 (SeparableConv (None, 16, 16, 728) 536536
['block7_sepconv3_act[0][0]']
2D)

block7_sepconv3_bn (BatchNorma (None, 16, 16, 728) 2912
['block7_sepconv3[0][0]']
lization)

add_5 (Add) (None, 16, 16, 728) 0
['block7_sepconv3_bn[0][0]',
'add_4[0][0]']

block8_sepconv1_act (Activatio (None, 16, 16, 728) 0 ['add_5[0][0]']
n)

block8_sepconv1 (SeparableConv (None, 16, 16, 728) 536536
['block8_sepconv1_act[0][0]']
2D)

block8_sepconv1_bn (BatchNorma (None, 16, 16, 728) 2912
['block8_sepconv1[0][0]']
lization)

block8_sepconv2_act (Activatio (None, 16, 16, 728) 0
['block8_sepconv1_bn[0][0]']
n)

block8_sepconv2 (SeparableConv (None, 16, 16, 728) 536536
['block8_sepconv2_act[0][0]']
2D)

block8_sepconv2_bn (BatchNorma (None, 16, 16, 728) 2912
['block8_sepconv2[0][0]']
lization)

block8_sepconv3_act (Activatio (None, 16, 16, 728) 0
['block8_sepconv2_bn[0][0]']
n)

block8_sepconv3 (SeparableConv (None, 16, 16, 728) 536536

```

```

['block8_sepconv3_act[0][0]']
2D)

block8_sepconv3_bn (BatchNorma (None, 16, 16, 728) 2912
['block8_sepconv3[0][0]']
lization)

add_6 (Add) (None, 16, 16, 728) 0
['block8_sepconv3_bn[0][0]',
'add_5[0][0]']

block9_sepconv1_act (Activatio (None, 16, 16, 728) 0 ['add_6[0][0]']
n)

block9_sepconv1 (SeparableConv (None, 16, 16, 728) 536536
['block9_sepconv1_act[0][0]']
2D)

block9_sepconv1_bn (BatchNorma (None, 16, 16, 728) 2912
['block9_sepconv1[0][0]']
lization)

block9_sepconv2_act (Activatio (None, 16, 16, 728) 0
['block9_sepconv1_bn[0][0]']
n)

block9_sepconv2 (SeparableConv (None, 16, 16, 728) 536536
['block9_sepconv2_act[0][0]']
2D)

block9_sepconv2_bn (BatchNorma (None, 16, 16, 728) 2912
['block9_sepconv2[0][0]']
lization)

block9_sepconv3_act (Activatio (None, 16, 16, 728) 0
['block9_sepconv2_bn[0][0]']
n)

block9_sepconv3 (SeparableConv (None, 16, 16, 728) 536536
['block9_sepconv3_act[0][0]']
2D)

block9_sepconv3_bn (BatchNorma (None, 16, 16, 728) 2912
['block9_sepconv3[0][0]']
lization)

add_7 (Add) (None, 16, 16, 728) 0
['block9_sepconv3_bn[0][0]',

```


			'add_6[0][0]'
block10_sepconv1_act (Activation)	(None, 16, 16, 728)	0	['add_7[0][0]']
block10_sepconv1 (SeparableConv2D)	(None, 16, 16, 728)	536536	
block10_sepconv1_bn (BatchNormalization)	(None, 16, 16, 728)	2912	
block10_sepconv2_act (Activation)	(None, 16, 16, 728)	0	
block10_sepconv2 (SeparableConv2D)	(None, 16, 16, 728)	536536	
block10_sepconv2_bn (BatchNormalization)	(None, 16, 16, 728)	2912	
block10_sepconv3_act (Activation)	(None, 16, 16, 728)	0	
block10_sepconv3 (SeparableConv2D)	(None, 16, 16, 728)	536536	
block10_sepconv3_bn (BatchNormalization)	(None, 16, 16, 728)	2912	
add_8 (Add)	(None, 16, 16, 728)	0	
			'add_7[0][0]'
block11_sepconv1_act (Activation)	(None, 16, 16, 728)	0	['add_8[0][0]']
block11_sepconv1 (SeparableConv2D)	(None, 16, 16, 728)	536536	

```

block11_sepconv1_bn (BatchNorm (None, 16, 16, 728) 2912
['block11_sepconv1[0][0]']
alization)

block11_sepconv2_act (Activati (None, 16, 16, 728) 0
['block11_sepconv1_bn[0][0]']
on)

block11_sepconv2 (SeparableCon (None, 16, 16, 728) 536536
['block11_sepconv2_act[0][0]']
v2D)

block11_sepconv2_bn (BatchNorm (None, 16, 16, 728) 2912
['block11_sepconv2[0][0]']
alization)

block11_sepconv3_act (Activati (None, 16, 16, 728) 0
['block11_sepconv2_bn[0][0]']
on)

block11_sepconv3 (SeparableCon (None, 16, 16, 728) 536536
['block11_sepconv3_act[0][0]']
v2D)

block11_sepconv3_bn (BatchNorm (None, 16, 16, 728) 2912
['block11_sepconv3[0][0]']
alization)

add_9 (Add) (None, 16, 16, 728) 0
['block11_sepconv3_bn[0][0]',
'add_8[0][0]']

block12_sepconv1_act (Activati (None, 16, 16, 728) 0 ['add_9[0][0]']
on)

block12_sepconv1 (SeparableCon (None, 16, 16, 728) 536536
['block12_sepconv1_act[0][0]']
v2D)

block12_sepconv1_bn (BatchNorm (None, 16, 16, 728) 2912
['block12_sepconv1[0][0]']
alization)

block12_sepconv2_act (Activati (None, 16, 16, 728) 0
['block12_sepconv1_bn[0][0]']
on)

block12_sepconv2 (SeparableCon (None, 16, 16, 728) 536536

```

```

['block12_sepconv2_act[0][0]']
v2D)

block12_sepconv2_bn (BatchNorm (None, 16, 16, 728) 2912
['block12_sepconv2[0][0]']
alization)

block12_sepconv3_act (Activati (None, 16, 16, 728) 0
['block12_sepconv2_bn[0][0]']
on)

block12_sepconv3 (SeparableCon (None, 16, 16, 728) 536536
['block12_sepconv3_act[0][0]']
v2D)

block12_sepconv3_bn (BatchNorm (None, 16, 16, 728) 2912
['block12_sepconv3[0][0]']
alization)

add_10 (Add) (None, 16, 16, 728) 0
['block12_sepconv3_bn[0][0]',
'add_9[0][0]']

block13_sepconv1_act (Activati (None, 16, 16, 728) 0
['add_10[0][0]']
on)

block13_sepconv1 (SeparableCon (None, 16, 16, 728) 536536
['block13_sepconv1_act[0][0]']
v2D)

block13_sepconv1_bn (BatchNorm (None, 16, 16, 728) 2912
['block13_sepconv1[0][0]']
alization)

block13_sepconv2_act (Activati (None, 16, 16, 728) 0
['block13_sepconv1_bn[0][0]']
on)

block13_sepconv2 (SeparableCon (None, 16, 16, 1024 752024
['block13_sepconv2_act[0][0]']
v2D) )

block13_sepconv2_bn (BatchNorm (None, 16, 16, 1024 4096
['block13_sepconv2[0][0]']
alization) )

conv2d_19 (Conv2D) (None, 8, 8, 1024) 745472

```

```

['add_10[0][0]']

block13_pool (MaxPooling2D)      (None, 8, 8, 1024)    0
['block13_sepconv2_bn[0][0]']

batch_normalization_19 (BatchN   (None, 8, 8, 1024)    4096
['conv2d_19[0][0]']
ormalization)

add_11 (Add)                      (None, 8, 8, 1024)    0
['block13_pool[0][0]',
'batch_normalization_19[0][0]']

block14_sepconv1 (SeparableCon   (None, 8, 8, 1536)    1582080
['add_11[0][0]']
v2D)

block14_sepconv1_bn (BatchNorm   (None, 8, 8, 1536)    6144
['block14_sepconv1[0][0]']
alization)

block14_sepconv1_act (Activati   (None, 8, 8, 1536)    0
['block14_sepconv1_bn[0][0]']
on)

block14_sepconv2 (SeparableCon   (None, 8, 8, 2048)    3159552
['block14_sepconv1_act[0][0]']
v2D)

block14_sepconv2_bn (BatchNorm   (None, 8, 8, 2048)    8192
['block14_sepconv2[0][0]']
alization)

block14_sepconv2_act (Activati   (None, 8, 8, 2048)    0
['block14_sepconv2_bn[0][0]']
on)

```

```

=====
=====
Total params: 20,861,480
Trainable params: 20,806,952
Non-trainable params: 54,528
-----
-----

```

```
[ ]: print(len(model.layers), model.layers[-1])
```

```
132 <keras.layers.core.activation.Activation object at 0x2cab0b9a0>
```

```
[ ]: model.trainable = False
      for layer in model.layers:
          assert not layer.trainable
```

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

Model: "xception"

```
-----
-----
Layer (type)                Output Shape              Param #   Connected to
=====
input_1 (InputLayer)        [(None, 256, 256, 3, 0
                           )]

block1_conv1 (Conv2D)        (None, 127, 127, 32, 864
['input_1[0][0]'])

block1_conv1_bn (BatchNormaliz (None, 127, 127, 32, 128
['block1_conv1[0][0]']
ation)

block1_conv1_act (Activation) (None, 127, 127, 32, 0
['block1_conv1_bn[0][0]'])

block1_conv2 (Conv2D)        (None, 125, 125, 64, 18432
['block1_conv1_act[0][0]'])

block1_conv2_bn (BatchNormaliz (None, 125, 125, 64, 256
['block1_conv2[0][0]']
ation)

block1_conv2_act (Activation) (None, 125, 125, 64, 0
['block1_conv2_bn[0][0]'])

block2_sepconv1 (SeparableConv (None, 125, 125, 12, 8768
['block1_conv2_act[0][0]']
2D)

block2_sepconv1_bn (BatchNorma (None, 125, 125, 12, 512
['block2_sepconv1[0][0]']
lization)

block2_sepconv2_act (Activatio (None, 125, 125, 12, 0
```

```

['block2_sepconv1_bn[0][0]']
n)                                8)

block2_sepconv2 (SeparableConv (None, 125, 125, 12 17536
['block2_sepconv2_act[0][0]']
2D)                                8)

block2_sepconv2_bn (BatchNorma (None, 125, 125, 12 512
['block2_sepconv2[0][0]']
lization)                        8)

conv2d_16 (Conv2D)                (None, 63, 63, 128) 8192
['block1_conv2_act[0][0]']

block2_pool (MaxPooling2D)        (None, 63, 63, 128) 0
['block2_sepconv2_bn[0][0]']

batch_normalization_16 (BatchN (None, 63, 63, 128) 512
['conv2d_16[0][0]']
ormalization)

add (Add)                        (None, 63, 63, 128) 0
['block2_pool[0][0]',
'batch_normalization_16[0][0]']

block3_sepconv1_act (Activatio (None, 63, 63, 128) 0      ['add[0][0]']
n)

block3_sepconv1 (SeparableConv (None, 63, 63, 256) 33920
['block3_sepconv1_act[0][0]']
2D)

block3_sepconv1_bn (BatchNorma (None, 63, 63, 256) 1024
['block3_sepconv1[0][0]']
lization)

block3_sepconv2_act (Activatio (None, 63, 63, 256) 0
['block3_sepconv1_bn[0][0]']
n)

block3_sepconv2 (SeparableConv (None, 63, 63, 256) 67840
['block3_sepconv2_act[0][0]']
2D)

block3_sepconv2_bn (BatchNorma (None, 63, 63, 256) 1024
['block3_sepconv2[0][0]']
lization)

```

conv2d_17 (Conv2D)	(None, 32, 32, 256)	32768	['add[0][0]']
block3_pool (MaxPooling2D)	(None, 32, 32, 256)	0	
['block3_sepconv2_bn[0][0]']			
batch_normalization_17 (Batch Normalization)	(None, 32, 32, 256)	1024	
['conv2d_17[0][0]']			
add_1 (Add)	(None, 32, 32, 256)	0	
['block3_pool[0][0]', 'batch_normalization_17[0][0]']			
block4_sepconv1_act (Activation)	(None, 32, 32, 256)	0	['add_1[0][0]']
block4_sepconv1 (SeparableConv2D)	(None, 32, 32, 728)	188672	
['block4_sepconv1_act[0][0]']			
block4_sepconv1_bn (Batch Normalization)	(None, 32, 32, 728)	2912	
['block4_sepconv1[0][0]']			
block4_sepconv2_act (Activation)	(None, 32, 32, 728)	0	
['block4_sepconv1_bn[0][0]']			
block4_sepconv2 (SeparableConv2D)	(None, 32, 32, 728)	536536	
['block4_sepconv2_act[0][0]']			
block4_sepconv2_bn (Batch Normalization)	(None, 32, 32, 728)	2912	
['block4_sepconv2[0][0]']			
conv2d_18 (Conv2D)	(None, 16, 16, 728)	186368	['add_1[0][0]']
block4_pool (MaxPooling2D)	(None, 16, 16, 728)	0	
['block4_sepconv2_bn[0][0]']			
batch_normalization_18 (Batch Normalization)	(None, 16, 16, 728)	2912	
['conv2d_18[0][0]']			
add_2 (Add)	(None, 16, 16, 728)	0	
['block4_pool[0][0]', 'batch_normalization_18[0][0]']			

block5_sepconv1_act (Activation)	(None, 16, 16, 728)	0	['add_2[0][0]']
block5_sepconv1 (SeparableConv2D)	(None, 16, 16, 728)	536536	['block5_sepconv1_act[0][0]']
block5_sepconv1_bn (BatchNormalization)	(None, 16, 16, 728)	2912	['block5_sepconv1[0][0]']
block5_sepconv2_act (Activation)	(None, 16, 16, 728)	0	['block5_sepconv1_bn[0][0]']
block5_sepconv2 (SeparableConv2D)	(None, 16, 16, 728)	536536	['block5_sepconv2_act[0][0]']
block5_sepconv2_bn (BatchNormalization)	(None, 16, 16, 728)	2912	['block5_sepconv2[0][0]']
block5_sepconv3_act (Activation)	(None, 16, 16, 728)	0	['block5_sepconv2_bn[0][0]']
block5_sepconv3 (SeparableConv2D)	(None, 16, 16, 728)	536536	['block5_sepconv3_act[0][0]']
block5_sepconv3_bn (BatchNormalization)	(None, 16, 16, 728)	2912	['block5_sepconv3[0][0]']
add_3 (Add)	(None, 16, 16, 728)	0	['block5_sepconv3_bn[0][0]', 'add_2[0][0]']
block6_sepconv1_act (Activation)	(None, 16, 16, 728)	0	['add_3[0][0]']
block6_sepconv1 (SeparableConv2D)	(None, 16, 16, 728)	536536	['block6_sepconv1_act[0][0]']
block6_sepconv1_bn (BatchNormalization)	(None, 16, 16, 728)	2912	


```

['block6_sepconv1[0][0]']
lization)

block6_sepconv2_act (Activation) (None, 16, 16, 728) 0
['block6_sepconv1_bn[0][0]']
n)

block6_sepconv2 (SeparableConv) (None, 16, 16, 728) 536536
['block6_sepconv2_act[0][0]']
2D)

block6_sepconv2_bn (BatchNormalization) (None, 16, 16, 728) 2912
['block6_sepconv2[0][0]']
lization)

block6_sepconv3_act (Activation) (None, 16, 16, 728) 0
['block6_sepconv2_bn[0][0]']
n)

block6_sepconv3 (SeparableConv) (None, 16, 16, 728) 536536
['block6_sepconv3_act[0][0]']
2D)

block6_sepconv3_bn (BatchNormalization) (None, 16, 16, 728) 2912
['block6_sepconv3[0][0]']
lization)

add_4 (Add) (None, 16, 16, 728) 0
['block6_sepconv3_bn[0][0]',
'add_3[0][0]']

block7_sepconv1_act (Activation) (None, 16, 16, 728) 0 ['add_4[0][0]']
n)

block7_sepconv1 (SeparableConv) (None, 16, 16, 728) 536536
['block7_sepconv1_act[0][0]']
2D)

block7_sepconv1_bn (BatchNormalization) (None, 16, 16, 728) 2912
['block7_sepconv1[0][0]']
lization)

block7_sepconv2_act (Activation) (None, 16, 16, 728) 0
['block7_sepconv1_bn[0][0]']
n)

block7_sepconv2 (SeparableConv) (None, 16, 16, 728) 536536
['block7_sepconv2_act[0][0]']

```

2D)

block7_sepconv2_bn (BatchNorma (None, 16, 16, 728) 2912
['block7_sepconv2[0][0]']
lization)

block7_sepconv3_act (Activatio (None, 16, 16, 728) 0
['block7_sepconv2_bn[0][0]']
n)

block7_sepconv3 (SeparableConv (None, 16, 16, 728) 536536
['block7_sepconv3_act[0][0]']
2D)

block7_sepconv3_bn (BatchNorma (None, 16, 16, 728) 2912
['block7_sepconv3[0][0]']
lization)

add_5 (Add) (None, 16, 16, 728) 0
['block7_sepconv3_bn[0][0]',

'add_4[0][0]']

block8_sepconv1_act (Activatio (None, 16, 16, 728) 0 ['add_5[0][0]']
n)

block8_sepconv1 (SeparableConv (None, 16, 16, 728) 536536
['block8_sepconv1_act[0][0]']
2D)

block8_sepconv1_bn (BatchNorma (None, 16, 16, 728) 2912
['block8_sepconv1[0][0]']
lization)

block8_sepconv2_act (Activatio (None, 16, 16, 728) 0
['block8_sepconv1_bn[0][0]']
n)

block8_sepconv2 (SeparableConv (None, 16, 16, 728) 536536
['block8_sepconv2_act[0][0]']
2D)

block8_sepconv2_bn (BatchNorma (None, 16, 16, 728) 2912
['block8_sepconv2[0][0]']
lization)

block8_sepconv3_act (Activatio (None, 16, 16, 728) 0
['block8_sepconv2_bn[0][0]']
n)

```

block8_sepconv3 (SeparableConv (None, 16, 16, 728) 536536
['block8_sepconv3_act[0][0]']
2D)

block8_sepconv3_bn (BatchNorma (None, 16, 16, 728) 2912
['block8_sepconv3[0][0]']
lization)

add_6 (Add) (None, 16, 16, 728) 0
['block8_sepconv3_bn[0][0]',
'add_5[0][0]']

block9_sepconv1_act (Activatio (None, 16, 16, 728) 0 ['add_6[0][0]']
n)

block9_sepconv1 (SeparableConv (None, 16, 16, 728) 536536
['block9_sepconv1_act[0][0]']
2D)

block9_sepconv1_bn (BatchNorma (None, 16, 16, 728) 2912
['block9_sepconv1[0][0]']
lization)

block9_sepconv2_act (Activatio (None, 16, 16, 728) 0
['block9_sepconv1_bn[0][0]']
n)

block9_sepconv2 (SeparableConv (None, 16, 16, 728) 536536
['block9_sepconv2_act[0][0]']
2D)

block9_sepconv2_bn (BatchNorma (None, 16, 16, 728) 2912
['block9_sepconv2[0][0]']
lization)

block9_sepconv3_act (Activatio (None, 16, 16, 728) 0
['block9_sepconv2_bn[0][0]']
n)

block9_sepconv3 (SeparableConv (None, 16, 16, 728) 536536
['block9_sepconv3_act[0][0]']
2D)

block9_sepconv3_bn (BatchNorma (None, 16, 16, 728) 2912
['block9_sepconv3[0][0]']
lization)

```

add_7 (Add)	(None, 16, 16, 728)	0	
['block9_sepconv3_bn[0][0]',			'add_6[0][0]']
block10_sepconv1_act (Activation)	(None, 16, 16, 728)	0	['add_7[0][0]']
block10_sepconv1 (SeparableConv2D)	(None, 16, 16, 728)	536536	
['block10_sepconv1_act[0][0]']			
block10_sepconv1_bn (BatchNormalization)	(None, 16, 16, 728)	2912	
['block10_sepconv1[0][0]']			
block10_sepconv2_act (Activation)	(None, 16, 16, 728)	0	
['block10_sepconv1_bn[0][0]']			
block10_sepconv2 (SeparableConv2D)	(None, 16, 16, 728)	536536	
['block10_sepconv2_act[0][0]']			
block10_sepconv2_bn (BatchNormalization)	(None, 16, 16, 728)	2912	
['block10_sepconv2[0][0]']			
block10_sepconv3_act (Activation)	(None, 16, 16, 728)	0	
['block10_sepconv2_bn[0][0]']			
block10_sepconv3 (SeparableConv2D)	(None, 16, 16, 728)	536536	
['block10_sepconv3_act[0][0]']			
block10_sepconv3_bn (BatchNormalization)	(None, 16, 16, 728)	2912	
['block10_sepconv3[0][0]']			
add_8 (Add)	(None, 16, 16, 728)	0	
['block10_sepconv3_bn[0][0]',			'add_7[0][0]']
block11_sepconv1_act (Activation)	(None, 16, 16, 728)	0	['add_8[0][0]']
block11_sepconv1 (SeparableConv2D)	(None, 16, 16, 728)	536536	
['block11_sepconv1_act[0][0]']			

```

v2D)

block11_sepconv1_bn (BatchNorm (None, 16, 16, 728) 2912
['block11_sepconv1[0][0]']
alization)

block11_sepconv2_act (Activati (None, 16, 16, 728) 0
['block11_sepconv1_bn[0][0]']
on)

block11_sepconv2 (SeparableCon (None, 16, 16, 728) 536536
['block11_sepconv2_act[0][0]']
v2D)

block11_sepconv2_bn (BatchNorm (None, 16, 16, 728) 2912
['block11_sepconv2[0][0]']
alization)

block11_sepconv3_act (Activati (None, 16, 16, 728) 0
['block11_sepconv2_bn[0][0]']
on)

block11_sepconv3 (SeparableCon (None, 16, 16, 728) 536536
['block11_sepconv3_act[0][0]']
v2D)

block11_sepconv3_bn (BatchNorm (None, 16, 16, 728) 2912
['block11_sepconv3[0][0]']
alization)

add_9 (Add) (None, 16, 16, 728) 0
['block11_sepconv3_bn[0][0]',
'add_8[0][0]']

block12_sepconv1_act (Activati (None, 16, 16, 728) 0 ['add_9[0][0]']
on)

block12_sepconv1 (SeparableCon (None, 16, 16, 728) 536536
['block12_sepconv1_act[0][0]']
v2D)

block12_sepconv1_bn (BatchNorm (None, 16, 16, 728) 2912
['block12_sepconv1[0][0]']
alization)

block12_sepconv2_act (Activati (None, 16, 16, 728) 0
['block12_sepconv1_bn[0][0]']
on)

```

```

block12_sepconv2 (SeparableCon (None, 16, 16, 728) 536536
['block12_sepconv2_act[0][0]']
v2D)

block12_sepconv2_bn (BatchNorm (None, 16, 16, 728) 2912
['block12_sepconv2[0][0]']
alization)

block12_sepconv3_act (Activati (None, 16, 16, 728) 0
['block12_sepconv2_bn[0][0]']
on)

block12_sepconv3 (SeparableCon (None, 16, 16, 728) 536536
['block12_sepconv3_act[0][0]']
v2D)

block12_sepconv3_bn (BatchNorm (None, 16, 16, 728) 2912
['block12_sepconv3[0][0]']
alization)

add_10 (Add) (None, 16, 16, 728) 0
['block12_sepconv3_bn[0][0]',
'add_9[0][0]']

block13_sepconv1_act (Activati (None, 16, 16, 728) 0
['add_10[0][0]']
on)

block13_sepconv1 (SeparableCon (None, 16, 16, 728) 536536
['block13_sepconv1_act[0][0]']
v2D)

block13_sepconv1_bn (BatchNorm (None, 16, 16, 728) 2912
['block13_sepconv1[0][0]']
alization)

block13_sepconv2_act (Activati (None, 16, 16, 728) 0
['block13_sepconv1_bn[0][0]']
on)

block13_sepconv2 (SeparableCon (None, 16, 16, 1024 752024
['block13_sepconv2_act[0][0]']
v2D) )

block13_sepconv2_bn (BatchNorm (None, 16, 16, 1024 4096
['block13_sepconv2[0][0]']
alization) )

```

```

conv2d_19 (Conv2D)          (None, 8, 8, 1024)  745472
['add_10[0][0]']

block13_pool (MaxPooling2D) (None, 8, 8, 1024)  0
['block13_sepconv2_bn[0][0]']

batch_normalization_19 (BatchN (None, 8, 8, 1024)  4096
['conv2d_19[0][0]']
ormalization)

add_11 (Add)                (None, 8, 8, 1024)  0
['block13_pool[0][0]',
'batch_normalization_19[0][0]']

block14_sepconv1 (SeparableCon (None, 8, 8, 1536)  1582080
['add_11[0][0]']
v2D)

block14_sepconv1_bn (BatchNorm (None, 8, 8, 1536)  6144
['block14_sepconv1[0][0]']
alization)

block14_sepconv1_act (Activati (None, 8, 8, 1536)  0
['block14_sepconv1_bn[0][0]']
on)

block14_sepconv2 (SeparableCon (None, 8, 8, 2048)  3159552
['block14_sepconv1_act[0][0]']
v2D)

block14_sepconv2_bn (BatchNorm (None, 8, 8, 2048)  8192
['block14_sepconv2[0][0]']
alization)

block14_sepconv2_act (Activati (None, 8, 8, 2048)  0
['block14_sepconv2_bn[0][0]']
on)

```

```

=====
=====

```

```

Total params: 20,861,480
Trainable params: 0
Non-trainable params: 20,861,480

```

```

-----
-----

```

Model jest dopasowany do wejścia 256x256x3, dodatkowo wszystko powyżej GAP jest usunięte oraz

warstwy zamrozone.

```
[ ]: base_model = model
      output = tf.keras.layers.GlobalAveragePooling2D()(base_model.output)
      output = tf.keras.layers.Dense(units=3, activation='softmax')(output)
      model = tf.keras.Model(base_model.input, output)
```

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

Model: "model"

```
-----
Layer (type)                 Output Shape              Param #   Connected to
=====
input_1 (InputLayer)         [(None, 256, 256, 3      0
)]

block1_conv1 (Conv2D)        (None, 127, 127, 32     864
['input_1[0][0]'])

block1_conv1_bn (BatchNormaliz (None, 127, 127, 32     128
['block1_conv1[0][0]']
ation)

block1_conv1_act (Activation) (None, 127, 127, 32     0
['block1_conv1_bn[0][0]'])

block1_conv2 (Conv2D)        (None, 125, 125, 64    18432
['block1_conv1_act[0][0]'])

block1_conv2_bn (BatchNormaliz (None, 125, 125, 64     256
['block1_conv2[0][0]']
ation)

block1_conv2_act (Activation) (None, 125, 125, 64     0
['block1_conv2_bn[0][0]'])

block2_sepconv1 (SeparableConv (None, 125, 125, 12     8768
2D)

block2_sepconv1_bn (BatchNorma (None, 125, 125, 12     512
['block2_sepconv1[0][0]'])
```



```

lization)                                8)

block2_sepconv2_act (Activatio (None, 125, 125, 12  0
['block2_sepconv1_bn[0][0]']
n)                                           8)

block2_sepconv2 (SeparableConv (None, 125, 125, 12  17536
['block2_sepconv2_act[0][0]']
2D)                                         8)

block2_sepconv2_bn (BatchNorma (None, 125, 125, 12  512
['block2_sepconv2[0][0]']
lization)                                8)

conv2d_16 (Conv2D)                    (None, 63, 63, 128) 8192
['block1_conv2_act[0][0]']

block2_pool (MaxPooling2D)            (None, 63, 63, 128) 0
['block2_sepconv2_bn[0][0]']

batch_normalization_16 (BatchN (None, 63, 63, 128) 512
['conv2d_16[0][0]']
ormalization)

add (Add)                             (None, 63, 63, 128) 0
['block2_pool[0][0]',
'batch_normalization_16[0][0]']

block3_sepconv1_act (Activatio (None, 63, 63, 128) 0      ['add[0][0]']
n)

block3_sepconv1 (SeparableConv (None, 63, 63, 256) 33920
['block3_sepconv1_act[0][0]']
2D)

block3_sepconv1_bn (BatchNorma (None, 63, 63, 256) 1024
['block3_sepconv1[0][0]']
lization)

block3_sepconv2_act (Activatio (None, 63, 63, 256) 0
['block3_sepconv1_bn[0][0]']
n)

block3_sepconv2 (SeparableConv (None, 63, 63, 256) 67840
['block3_sepconv2_act[0][0]']
2D)

block3_sepconv2_bn (BatchNorma (None, 63, 63, 256) 1024

```

```

['block3_sepconv2[0][0]']
lization)

conv2d_17 (Conv2D)          (None, 32, 32, 256) 32768      ['add[0][0]']

block3_pool (MaxPooling2D)  (None, 32, 32, 256) 0
['block3_sepconv2_bn[0][0]']

batch_normalization_17 (BatchN (None, 32, 32, 256) 1024
['conv2d_17[0][0]']
ormalization)

add_1 (Add)                 (None, 32, 32, 256) 0
['block3_pool[0][0]',
'batch_normalization_17[0][0]']

block4_sepconv1_act (Activatio (None, 32, 32, 256) 0      ['add_1[0][0]']
n)

block4_sepconv1 (SeparableConv (None, 32, 32, 728) 188672
['block4_sepconv1_act[0][0]']
2D)

block4_sepconv1_bn (BatchNorma (None, 32, 32, 728) 2912
['block4_sepconv1[0][0]']
lization)

block4_sepconv2_act (Activatio (None, 32, 32, 728) 0
['block4_sepconv1_bn[0][0]']
n)

block4_sepconv2 (SeparableConv (None, 32, 32, 728) 536536
['block4_sepconv2_act[0][0]']
2D)

block4_sepconv2_bn (BatchNorma (None, 32, 32, 728) 2912
['block4_sepconv2[0][0]']
lization)

conv2d_18 (Conv2D)          (None, 16, 16, 728) 186368      ['add_1[0][0]']

block4_pool (MaxPooling2D)    (None, 16, 16, 728) 0
['block4_sepconv2_bn[0][0]']

batch_normalization_18 (BatchN (None, 16, 16, 728) 2912
['conv2d_18[0][0]']
ormalization)

```

```

add_2 (Add) (None, 16, 16, 728) 0
['block4_pool[0][0]',
'batch_normalization_18[0][0]']

block5_sepconv1_act (Activation) (None, 16, 16, 728) 0 ['add_2[0][0]']

block5_sepconv1 (SeparableConv2D) (None, 16, 16, 728) 536536
['block5_sepconv1_act[0][0]']

block5_sepconv1_bn (BatchNormalization) (None, 16, 16, 728) 2912
['block5_sepconv1[0][0]']

block5_sepconv2_act (Activation) (None, 16, 16, 728) 0
['block5_sepconv1_bn[0][0]']

block5_sepconv2 (SeparableConv2D) (None, 16, 16, 728) 536536
['block5_sepconv2_act[0][0]']

block5_sepconv2_bn (BatchNormalization) (None, 16, 16, 728) 2912
['block5_sepconv2[0][0]']

block5_sepconv3_act (Activation) (None, 16, 16, 728) 0
['block5_sepconv2_bn[0][0]']

block5_sepconv3 (SeparableConv2D) (None, 16, 16, 728) 536536
['block5_sepconv3_act[0][0]']

block5_sepconv3_bn (BatchNormalization) (None, 16, 16, 728) 2912
['block5_sepconv3[0][0]']

add_3 (Add) (None, 16, 16, 728) 0
['block5_sepconv3_bn[0][0]',
'add_2[0][0]']

block6_sepconv1_act (Activation) (None, 16, 16, 728) 0 ['add_3[0][0]']

block6_sepconv1 (SeparableConv2D) (None, 16, 16, 728) 536536
['block6_sepconv1_act[0][0]']

```

2D)

block6_sepconv1_bn (BatchNorma (None, 16, 16, 728) 2912
['block6_sepconv1[0][0]']
lization)

block6_sepconv2_act (Activatio (None, 16, 16, 728) 0
['block6_sepconv1_bn[0][0]']
n)

block6_sepconv2 (SeparableConv (None, 16, 16, 728) 536536
['block6_sepconv2_act[0][0]']
2D)

block6_sepconv2_bn (BatchNorma (None, 16, 16, 728) 2912
['block6_sepconv2[0][0]']
lization)

block6_sepconv3_act (Activatio (None, 16, 16, 728) 0
['block6_sepconv2_bn[0][0]']
n)

block6_sepconv3 (SeparableConv (None, 16, 16, 728) 536536
['block6_sepconv3_act[0][0]']
2D)

block6_sepconv3_bn (BatchNorma (None, 16, 16, 728) 2912
['block6_sepconv3[0][0]']
lization)

add_4 (Add) (None, 16, 16, 728) 0
['block6_sepconv3_bn[0][0]',
'add_3[0][0]']

block7_sepconv1_act (Activatio (None, 16, 16, 728) 0 ['add_4[0][0]']
n)

block7_sepconv1 (SeparableConv (None, 16, 16, 728) 536536
['block7_sepconv1_act[0][0]']
2D)

block7_sepconv1_bn (BatchNorma (None, 16, 16, 728) 2912
['block7_sepconv1[0][0]']
lization)

block7_sepconv2_act (Activatio (None, 16, 16, 728) 0
['block7_sepconv1_bn[0][0]']
n)

```

block7_sepconv2 (SeparableConv (None, 16, 16, 728) 536536
['block7_sepconv2_act[0][0]']
2D)

block7_sepconv2_bn (BatchNorma (None, 16, 16, 728) 2912
['block7_sepconv2[0][0]']
lization)

block7_sepconv3_act (Activatio (None, 16, 16, 728) 0
['block7_sepconv2_bn[0][0]']
n)

block7_sepconv3 (SeparableConv (None, 16, 16, 728) 536536
['block7_sepconv3_act[0][0]']
2D)

block7_sepconv3_bn (BatchNorma (None, 16, 16, 728) 2912
['block7_sepconv3[0][0]']
lization)

add_5 (Add) (None, 16, 16, 728) 0
['block7_sepconv3_bn[0][0]',
'add_4[0][0]']

block8_sepconv1_act (Activatio (None, 16, 16, 728) 0 ['add_5[0][0]']
n)

block8_sepconv1 (SeparableConv (None, 16, 16, 728) 536536
['block8_sepconv1_act[0][0]']
2D)

block8_sepconv1_bn (BatchNorma (None, 16, 16, 728) 2912
['block8_sepconv1[0][0]']
lization)

block8_sepconv2_act (Activatio (None, 16, 16, 728) 0
['block8_sepconv1_bn[0][0]']
n)

block8_sepconv2 (SeparableConv (None, 16, 16, 728) 536536
['block8_sepconv2_act[0][0]']
2D)

block8_sepconv2_bn (BatchNorma (None, 16, 16, 728) 2912
['block8_sepconv2[0][0]']
lization)

```

```

block8_sepconv3_act (Activation) (None, 16, 16, 728) 0
['block8_sepconv2_bn[0][0]']
n)

block8_sepconv3 (SeparableConv) (None, 16, 16, 728) 536536
['block8_sepconv3_act[0][0]']
2D)

block8_sepconv3_bn (BatchNormalization) (None, 16, 16, 728) 2912
['block8_sepconv3[0][0]']
lization)

add_6 (Add) (None, 16, 16, 728) 0
['block8_sepconv3_bn[0][0]',
'add_5[0][0]']

block9_sepconv1_act (Activation) (None, 16, 16, 728) 0 ['add_6[0][0]']
n)

block9_sepconv1 (SeparableConv) (None, 16, 16, 728) 536536
['block9_sepconv1_act[0][0]']
2D)

block9_sepconv1_bn (BatchNormalization) (None, 16, 16, 728) 2912
['block9_sepconv1[0][0]']
lization)

block9_sepconv2_act (Activation) (None, 16, 16, 728) 0
['block9_sepconv1_bn[0][0]']
n)

block9_sepconv2 (SeparableConv) (None, 16, 16, 728) 536536
['block9_sepconv2_act[0][0]']
2D)

block9_sepconv2_bn (BatchNormalization) (None, 16, 16, 728) 2912
['block9_sepconv2[0][0]']
lization)

block9_sepconv3_act (Activation) (None, 16, 16, 728) 0
['block9_sepconv2_bn[0][0]']
n)

block9_sepconv3 (SeparableConv) (None, 16, 16, 728) 536536
['block9_sepconv3_act[0][0]']
2D)

block9_sepconv3_bn (BatchNormalization) (None, 16, 16, 728) 2912

```

```

['block9_sepconv3[0][0]']
lization)

add_7 (Add) (None, 16, 16, 728) 0
['block9_sepconv3_bn[0][0]',
'add_6[0][0]']

block10_sepconv1_act (Activati (None, 16, 16, 728) 0 ['add_7[0][0]']
on)

block10_sepconv1 (SeparableCon (None, 16, 16, 728) 536536
['block10_sepconv1_act[0][0]']
v2D)

block10_sepconv1_bn (BatchNorm (None, 16, 16, 728) 2912
['block10_sepconv1[0][0]']
alization)

block10_sepconv2_act (Activati (None, 16, 16, 728) 0
['block10_sepconv1_bn[0][0]']
on)

block10_sepconv2 (SeparableCon (None, 16, 16, 728) 536536
['block10_sepconv2_act[0][0]']
v2D)

block10_sepconv2_bn (BatchNorm (None, 16, 16, 728) 2912
['block10_sepconv2[0][0]']
alization)

block10_sepconv3_act (Activati (None, 16, 16, 728) 0
['block10_sepconv2_bn[0][0]']
on)

block10_sepconv3 (SeparableCon (None, 16, 16, 728) 536536
['block10_sepconv3_act[0][0]']
v2D)

block10_sepconv3_bn (BatchNorm (None, 16, 16, 728) 2912
['block10_sepconv3[0][0]']
alization)

add_8 (Add) (None, 16, 16, 728) 0
['block10_sepconv3_bn[0][0]',
'add_7[0][0]']

block11_sepconv1_act (Activati (None, 16, 16, 728) 0 ['add_8[0][0]']
on)

```

```

block11_sepconv1 (SeparableCon (None, 16, 16, 728) 536536
['block11_sepconv1_act[0][0]']
v2D)

block11_sepconv1_bn (BatchNorm (None, 16, 16, 728) 2912
['block11_sepconv1[0][0]']
alization)

block11_sepconv2_act (Activati (None, 16, 16, 728) 0
['block11_sepconv1_bn[0][0]']
on)

block11_sepconv2 (SeparableCon (None, 16, 16, 728) 536536
['block11_sepconv2_act[0][0]']
v2D)

block11_sepconv2_bn (BatchNorm (None, 16, 16, 728) 2912
['block11_sepconv2[0][0]']
alization)

block11_sepconv3_act (Activati (None, 16, 16, 728) 0
['block11_sepconv2_bn[0][0]']
on)

block11_sepconv3 (SeparableCon (None, 16, 16, 728) 536536
['block11_sepconv3_act[0][0]']
v2D)

block11_sepconv3_bn (BatchNorm (None, 16, 16, 728) 2912
['block11_sepconv3[0][0]']
alization)

add_9 (Add) (None, 16, 16, 728) 0
['block11_sepconv3_bn[0][0]',
'add_8[0][0]']

block12_sepconv1_act (Activati (None, 16, 16, 728) 0 ['add_9[0][0]']
on)

block12_sepconv1 (SeparableCon (None, 16, 16, 728) 536536
['block12_sepconv1_act[0][0]']
v2D)

block12_sepconv1_bn (BatchNorm (None, 16, 16, 728) 2912
['block12_sepconv1[0][0]']
alization)

```



```

    block12_sepconv2_act (Activation) (None, 16, 16, 728) 0
    ['block12_sepconv1_bn[0][0]']
    on)

    block12_sepconv2 (SeparableConv2D) (None, 16, 16, 728) 536536
    ['block12_sepconv2_act[0][0]']
    v2D)

    block12_sepconv2_bn (BatchNormalization) (None, 16, 16, 728) 2912
    ['block12_sepconv2[0][0]']
    alization)

    block12_sepconv3_act (Activation) (None, 16, 16, 728) 0
    ['block12_sepconv2_bn[0][0]']
    on)

    block12_sepconv3 (SeparableConv2D) (None, 16, 16, 728) 536536
    ['block12_sepconv3_act[0][0]']
    v2D)

    block12_sepconv3_bn (BatchNormalization) (None, 16, 16, 728) 2912
    ['block12_sepconv3[0][0]']
    alization)

    add_10 (Add) (None, 16, 16, 728) 0
    ['block12_sepconv3_bn[0][0]',
    'add_9[0][0]']

    block13_sepconv1_act (Activation) (None, 16, 16, 728) 0
    ['add_10[0][0]']
    on)

    block13_sepconv1 (SeparableConv2D) (None, 16, 16, 728) 536536
    ['block13_sepconv1_act[0][0]']
    v2D)

    block13_sepconv1_bn (BatchNormalization) (None, 16, 16, 728) 2912
    ['block13_sepconv1[0][0]']
    alization)

    block13_sepconv2_act (Activation) (None, 16, 16, 728) 0
    ['block13_sepconv1_bn[0][0]']
    on)

    block13_sepconv2 (SeparableConv2D) (None, 16, 16, 1024) 752024
    ['block13_sepconv2_act[0][0]']
    v2D)
    )

```

```

block13_sepconv2_bn (BatchNorm (None, 16, 16, 1024) 4096
['block13_sepconv2[0][0]']
alization)

conv2d_19 (Conv2D) (None, 8, 8, 1024) 745472
['add_10[0][0]']

block13_pool (MaxPooling2D) (None, 8, 8, 1024) 0
['block13_sepconv2_bn[0][0]']

batch_normalization_19 (BatchN (None, 8, 8, 1024) 4096
['conv2d_19[0][0]']
ormalization)

add_11 (Add) (None, 8, 8, 1024) 0
['block13_pool[0][0]',
'batch_normalization_19[0][0]']

block14_sepconv1 (SeparableCon (None, 8, 8, 1536) 1582080
['add_11[0][0]']
v2D)

block14_sepconv1_bn (BatchNorm (None, 8, 8, 1536) 6144
['block14_sepconv1[0][0]']
alization)

block14_sepconv1_act (Activati (None, 8, 8, 1536) 0
['block14_sepconv1_bn[0][0]']
on)

block14_sepconv2 (SeparableCon (None, 8, 8, 2048) 3159552
['block14_sepconv1_act[0][0]']
v2D)

block14_sepconv2_bn (BatchNorm (None, 8, 8, 2048) 8192
['block14_sepconv2[0][0]']
alization)

block14_sepconv2_act (Activati (None, 8, 8, 2048) 0
['block14_sepconv2_bn[0][0]']
on)

global_average_pooling2d_2 (Gl (None, 2048) 0
['block14_sepconv2_act[0][0]']
obalAveragePooling2D)

dense_3 (Dense) (None, 3) 6147
['global_average_pooling2d_2[0][0]']

```

']']

```
=====
=====
Total params: 20,867,627
Trainable params: 6,147
Non-trainable params: 20,861,480
-----
-----
```

Stworzyłem nowy model, który składa się z modelu xception, GAP i warstwy wyjściowej.

```
[ ]: for layer in model.layers[:-2]:
      assert not layer.trainable
assert model.layers[-2].trainable
assert model.layers[-1].trainable
```

```
[ ]: model.compile(
      optimizer=tf.keras.optimizers.SGD(learning_rate=0.001, momentum=0.9),
      loss=tf.keras.losses.CategoricalCrossentropy(),
      metrics=['accuracy']
)
```

```
[ ]: history = model.fit(
      x=train_ds,
      batch_size=64,
      epochs=30,
      verbose=1,
      validation_data=test_ds
)
```

Epoch 1/30

2022-11-04 00:28:57.022150: I
tensorflow/core/grappler/optimizers/custom_graph_optimizer_registry.cc:114]
Plugin optimizer for device_type GPU is enabled.

12/12 [=====] - ETA: 0s - loss: 1.0921 - accuracy:
0.3910

2022-11-04 00:29:04.254271: I
tensorflow/core/grappler/optimizers/custom_graph_optimizer_registry.cc:114]
Plugin optimizer for device_type GPU is enabled.

12/12 [=====] - 11s 793ms/step - loss: 1.0921 -
accuracy: 0.3910 - val_loss: 0.8689 - val_accuracy: 0.7037

Epoch 2/30

12/12 [=====] - 9s 748ms/step - loss: 0.6989 -
accuracy: 0.8230 - val_loss: 0.5309 - val_accuracy: 0.9312

Epoch 3/30

12/12 [=====] - 9s 707ms/step - loss: 0.4407 -

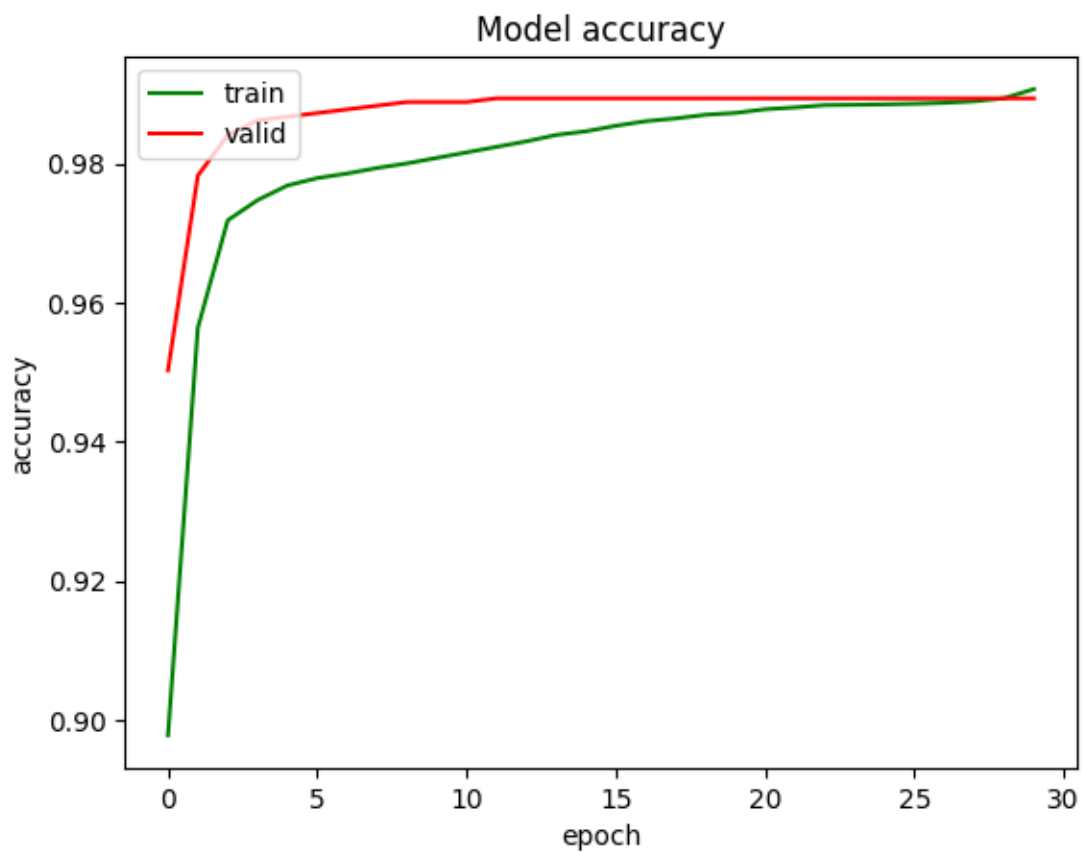
accuracy: 0.9498 - val_loss: 0.3696 - val_accuracy: 0.9683
Epoch 4/30
12/12 [=====] - 9s 710ms/step - loss: 0.3199 -
accuracy: 0.9617 - val_loss: 0.2901 - val_accuracy: 0.9841
Epoch 5/30
12/12 [=====] - 8s 642ms/step - loss: 0.2577 -
accuracy: 0.9696 - val_loss: 0.2441 - val_accuracy: 0.9841
Epoch 6/30
12/12 [=====] - 8s 640ms/step - loss: 0.2207 -
accuracy: 0.9749 - val_loss: 0.2149 - val_accuracy: 0.9841
Epoch 7/30
12/12 [=====] - 8s 642ms/step - loss: 0.1962 -
accuracy: 0.9762 - val_loss: 0.1949 - val_accuracy: 0.9841
Epoch 8/30
12/12 [=====] - 8s 642ms/step - loss: 0.1781 -
accuracy: 0.9762 - val_loss: 0.1790 - val_accuracy: 0.9841
Epoch 9/30
12/12 [=====] - 8s 641ms/step - loss: 0.1644 -
accuracy: 0.9775 - val_loss: 0.1663 - val_accuracy: 0.9894
Epoch 10/30
12/12 [=====] - 8s 640ms/step - loss: 0.1529 -
accuracy: 0.9775 - val_loss: 0.1563 - val_accuracy: 0.9894
Epoch 11/30
12/12 [=====] - 8s 642ms/step - loss: 0.1439 -
accuracy: 0.9775 - val_loss: 0.1482 - val_accuracy: 0.9841
Epoch 12/30
12/12 [=====] - 8s 641ms/step - loss: 0.1360 -
accuracy: 0.9775 - val_loss: 0.1408 - val_accuracy: 0.9894
Epoch 13/30
12/12 [=====] - 8s 640ms/step - loss: 0.1294 -
accuracy: 0.9789 - val_loss: 0.1342 - val_accuracy: 0.9894
Epoch 14/30
12/12 [=====] - 8s 639ms/step - loss: 0.1235 -
accuracy: 0.9828 - val_loss: 0.1287 - val_accuracy: 0.9894
Epoch 15/30
12/12 [=====] - 8s 640ms/step - loss: 0.1183 -
accuracy: 0.9802 - val_loss: 0.1241 - val_accuracy: 0.9894
Epoch 16/30
12/12 [=====] - 8s 641ms/step - loss: 0.1138 -
accuracy: 0.9815 - val_loss: 0.1198 - val_accuracy: 0.9894
Epoch 17/30
12/12 [=====] - 8s 640ms/step - loss: 0.1096 -
accuracy: 0.9841 - val_loss: 0.1158 - val_accuracy: 0.9894
Epoch 18/30
12/12 [=====] - 8s 641ms/step - loss: 0.1059 -
accuracy: 0.9828 - val_loss: 0.1121 - val_accuracy: 0.9894
Epoch 19/30
12/12 [=====] - 8s 640ms/step - loss: 0.1025 -

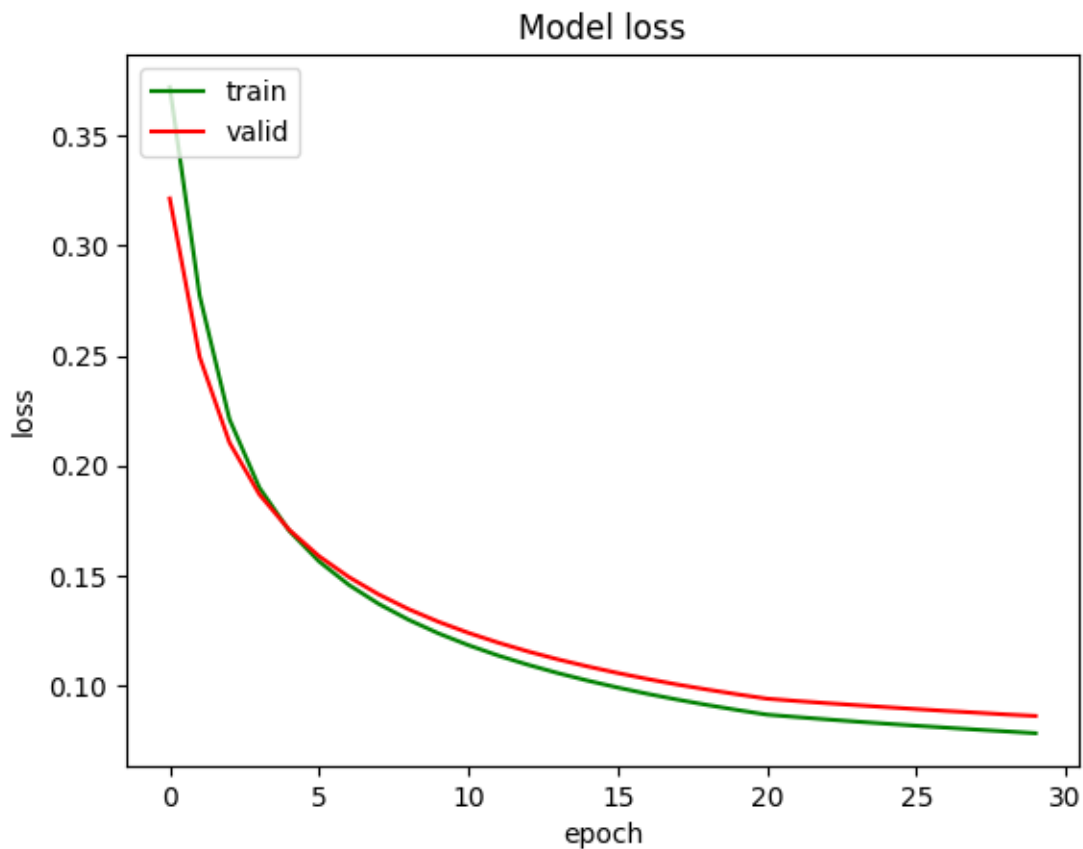
```

accuracy: 0.9855 - val_loss: 0.1088 - val_accuracy: 0.9894
Epoch 20/30
12/12 [=====] - 8s 641ms/step - loss: 0.0994 -
accuracy: 0.9855 - val_loss: 0.1061 - val_accuracy: 0.9894
Epoch 21/30
12/12 [=====] - 8s 641ms/step - loss: 0.0966 -
accuracy: 0.9855 - val_loss: 0.1035 - val_accuracy: 0.9894
Epoch 22/30
12/12 [=====] - 8s 642ms/step - loss: 0.0940 -
accuracy: 0.9855 - val_loss: 0.1007 - val_accuracy: 0.9894
Epoch 23/30
12/12 [=====] - 8s 641ms/step - loss: 0.0916 -
accuracy: 0.9881 - val_loss: 0.0982 - val_accuracy: 0.9894
Epoch 24/30
12/12 [=====] - 8s 641ms/step - loss: 0.0892 -
accuracy: 0.9881 - val_loss: 0.0964 - val_accuracy: 0.9894
Epoch 25/30
12/12 [=====] - 8s 641ms/step - loss: 0.0871 -
accuracy: 0.9881 - val_loss: 0.0944 - val_accuracy: 0.9894
Epoch 26/30
12/12 [=====] - 8s 641ms/step - loss: 0.0851 -
accuracy: 0.9881 - val_loss: 0.0925 - val_accuracy: 0.9894
Epoch 27/30
12/12 [=====] - 8s 641ms/step - loss: 0.0834 -
accuracy: 0.9881 - val_loss: 0.0906 - val_accuracy: 0.9894
Epoch 28/30
12/12 [=====] - 8s 640ms/step - loss: 0.0815 -
accuracy: 0.9881 - val_loss: 0.0892 - val_accuracy: 0.9894
Epoch 29/30
12/12 [=====] - 8s 641ms/step - loss: 0.0798 -
accuracy: 0.9881 - val_loss: 0.0874 - val_accuracy: 0.9894
Epoch 30/30
12/12 [=====] - 8s 641ms/step - loss: 0.0782 -
accuracy: 0.9908 - val_loss: 0.0860 - val_accuracy: 0.9894

```

```
[ ]: print_history(history.history)
```





```
[ ]: model.save_weights('./checkpoints/imagenet')
```

Tym razem otrzymaliśmy naprawdę ładne wyniki, co nie jest zaskakujące biorąc pod uwagę jak mocny model wykorzystujemy.

Co ciekawe accuracy na zbiorze walidacyjnym przez spory czas wychodziło większe niż na zbiorze treningowym.

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

Model: "model"

Layer (type)	Output Shape	Param #	Connected to
=====			
input_1 (InputLayer)	[(None, 256, 256, 3 0)]		[]
block1_conv1 (Conv2D)	(None, 127, 127, 32 864 ['input_1[0][0]']))

```

block1_conv1_bn (BatchNormaliz (None, 127, 127, 32 128
['block1_conv1[0][0]']
ation)
)

block1_conv1_act (Activation) (None, 127, 127, 32 0
['block1_conv1_bn[0][0]']
)

block1_conv2 (Conv2D) (None, 125, 125, 64 18432
['block1_conv1_act[0][0]']
)

block1_conv2_bn (BatchNormaliz (None, 125, 125, 64 256
['block1_conv2[0][0]']
ation)
)

block1_conv2_act (Activation) (None, 125, 125, 64 0
['block1_conv2_bn[0][0]']
)

block2_sepconv1 (SeparableConv (None, 125, 125, 12 8768
['block1_conv2_act[0][0]']
2D)
8)

block2_sepconv1_bn (BatchNorma (None, 125, 125, 12 512
['block2_sepconv1[0][0]']
lization)
8)

block2_sepconv2_act (Activatio (None, 125, 125, 12 0
['block2_sepconv1_bn[0][0]']
n)
8)

block2_sepconv2 (SeparableConv (None, 125, 125, 12 17536
['block2_sepconv2_act[0][0]']
2D)
8)

block2_sepconv2_bn (BatchNorma (None, 125, 125, 12 512
['block2_sepconv2[0][0]']
lization)
8)

conv2d_16 (Conv2D) (None, 63, 63, 128) 8192
['block1_conv2_act[0][0]']

block2_pool (MaxPooling2D) (None, 63, 63, 128) 0
['block2_sepconv2_bn[0][0]']

batch_normalization_16 (BatchN (None, 63, 63, 128) 512

```



```

['conv2d_16[0][0]']
ormalization)

add (Add) (None, 63, 63, 128) 0
['block2_pool[0][0]',
'batch_normalization_16[0][0]']

block3_sepconv1_act (Activation) (None, 63, 63, 128) 0 ['add[0][0]']

block3_sepconv1 (SeparableConv2D) (None, 63, 63, 256) 33920
['block3_sepconv1_act[0][0]']

block3_sepconv1_bn (BatchNormalization) (None, 63, 63, 256) 1024
['block3_sepconv1[0][0]']

block3_sepconv2_act (Activation) (None, 63, 63, 256) 0
['block3_sepconv1_bn[0][0]']

block3_sepconv2 (SeparableConv2D) (None, 63, 63, 256) 67840
['block3_sepconv2_act[0][0]']

block3_sepconv2_bn (BatchNormalization) (None, 63, 63, 256) 1024
['block3_sepconv2[0][0]']

conv2d_17 (Conv2D) (None, 32, 32, 256) 32768 ['add[0][0]']

block3_pool (MaxPooling2D) (None, 32, 32, 256) 0
['block3_sepconv2_bn[0][0]']

batch_normalization_17 (BatchNormalization) (None, 32, 32, 256) 1024
['conv2d_17[0][0]']

add_1 (Add) (None, 32, 32, 256) 0
['block3_pool[0][0]',
'batch_normalization_17[0][0]']

block4_sepconv1_act (Activation) (None, 32, 32, 256) 0 ['add_1[0][0]']

block4_sepconv1 (SeparableConv2D) (None, 32, 32, 728) 188672
['block4_sepconv1_act[0][0]']

```

```

2D)

block4_sepconv1_bn (BatchNorma (None, 32, 32, 728) 2912
['block4_sepconv1[0][0]']
lization)

block4_sepconv2_act (Activatio (None, 32, 32, 728) 0
['block4_sepconv1_bn[0][0]']
n)

block4_sepconv2 (SeparableConv (None, 32, 32, 728) 536536
['block4_sepconv2_act[0][0]']
2D)

block4_sepconv2_bn (BatchNorma (None, 32, 32, 728) 2912
['block4_sepconv2[0][0]']
lization)

conv2d_18 (Conv2D) (None, 16, 16, 728) 186368 ['add_1[0][0]']

block4_pool (MaxPooling2D) (None, 16, 16, 728) 0
['block4_sepconv2_bn[0][0]']

batch_normalization_18 (BatchN (None, 16, 16, 728) 2912
['conv2d_18[0][0]']
ormalization)

add_2 (Add) (None, 16, 16, 728) 0
['block4_pool[0][0]',
'batch_normalization_18[0][0]']

block5_sepconv1_act (Activatio (None, 16, 16, 728) 0 ['add_2[0][0]']
n)

block5_sepconv1 (SeparableConv (None, 16, 16, 728) 536536
['block5_sepconv1_act[0][0]']
2D)

block5_sepconv1_bn (BatchNorma (None, 16, 16, 728) 2912
['block5_sepconv1[0][0]']
lization)

block5_sepconv2_act (Activatio (None, 16, 16, 728) 0
['block5_sepconv1_bn[0][0]']
n)

block5_sepconv2 (SeparableConv (None, 16, 16, 728) 536536
['block5_sepconv2_act[0][0]']

```

2D)

block5_sepconv2_bn (BatchNorma (None, 16, 16, 728) 2912
['block5_sepconv2[0][0]']
lization)

block5_sepconv3_act (Activatio (None, 16, 16, 728) 0
['block5_sepconv2_bn[0][0]']
n)

block5_sepconv3 (SeparableConv (None, 16, 16, 728) 536536
['block5_sepconv3_act[0][0]']
2D)

block5_sepconv3_bn (BatchNorma (None, 16, 16, 728) 2912
['block5_sepconv3[0][0]']
lization)

add_3 (Add) (None, 16, 16, 728) 0
['block5_sepconv3_bn[0][0]',

'add_2[0][0]']

block6_sepconv1_act (Activatio (None, 16, 16, 728) 0 ['add_3[0][0]']
n)

block6_sepconv1 (SeparableConv (None, 16, 16, 728) 536536
['block6_sepconv1_act[0][0]']
2D)

block6_sepconv1_bn (BatchNorma (None, 16, 16, 728) 2912
['block6_sepconv1[0][0]']
lization)

block6_sepconv2_act (Activatio (None, 16, 16, 728) 0
['block6_sepconv1_bn[0][0]']
n)

block6_sepconv2 (SeparableConv (None, 16, 16, 728) 536536
['block6_sepconv2_act[0][0]']
2D)

block6_sepconv2_bn (BatchNorma (None, 16, 16, 728) 2912
['block6_sepconv2[0][0]']
lization)

block6_sepconv3_act (Activatio (None, 16, 16, 728) 0
['block6_sepconv2_bn[0][0]']
n)

```

block6_sepconv3 (SeparableConv (None, 16, 16, 728) 536536
['block6_sepconv3_act[0][0]']
2D)

block6_sepconv3_bn (BatchNorma (None, 16, 16, 728) 2912
['block6_sepconv3[0][0]']
lization)

add_4 (Add) (None, 16, 16, 728) 0
['block6_sepconv3_bn[0][0]',
'add_3[0][0]']

block7_sepconv1_act (Activatio (None, 16, 16, 728) 0 ['add_4[0][0]']
n)

block7_sepconv1 (SeparableConv (None, 16, 16, 728) 536536
['block7_sepconv1_act[0][0]']
2D)

block7_sepconv1_bn (BatchNorma (None, 16, 16, 728) 2912
['block7_sepconv1[0][0]']
lization)

block7_sepconv2_act (Activatio (None, 16, 16, 728) 0
['block7_sepconv1_bn[0][0]']
n)

block7_sepconv2 (SeparableConv (None, 16, 16, 728) 536536
['block7_sepconv2_act[0][0]']
2D)

block7_sepconv2_bn (BatchNorma (None, 16, 16, 728) 2912
['block7_sepconv2[0][0]']
lization)

block7_sepconv3_act (Activatio (None, 16, 16, 728) 0
['block7_sepconv2_bn[0][0]']
n)

block7_sepconv3 (SeparableConv (None, 16, 16, 728) 536536
['block7_sepconv3_act[0][0]']
2D)

block7_sepconv3_bn (BatchNorma (None, 16, 16, 728) 2912
['block7_sepconv3[0][0]']
lization)

```

add_5 (Add)	(None, 16, 16, 728)	0	
['block7_sepconv3_bn[0][0]',			'add_4[0][0]']
block8_sepconv1_act (Activation)	(None, 16, 16, 728)	0	['add_5[0][0]']
block8_sepconv1 (SeparableConv2D)	(None, 16, 16, 728)	536536	
['block8_sepconv1_act[0][0]']			
block8_sepconv1_bn (BatchNormalization)	(None, 16, 16, 728)	2912	
['block8_sepconv1[0][0]']			
block8_sepconv2_act (Activation)	(None, 16, 16, 728)	0	
['block8_sepconv1_bn[0][0]']			
block8_sepconv2 (SeparableConv2D)	(None, 16, 16, 728)	536536	
['block8_sepconv2_act[0][0]']			
block8_sepconv2_bn (BatchNormalization)	(None, 16, 16, 728)	2912	
['block8_sepconv2[0][0]']			
block8_sepconv3_act (Activation)	(None, 16, 16, 728)	0	
['block8_sepconv2_bn[0][0]']			
block8_sepconv3 (SeparableConv2D)	(None, 16, 16, 728)	536536	
['block8_sepconv3_act[0][0]']			
block8_sepconv3_bn (BatchNormalization)	(None, 16, 16, 728)	2912	
['block8_sepconv3[0][0]']			
add_6 (Add)	(None, 16, 16, 728)	0	
['block8_sepconv3_bn[0][0]',			'add_5[0][0]']
block9_sepconv1_act (Activation)	(None, 16, 16, 728)	0	['add_6[0][0]']
block9_sepconv1 (SeparableConv2D)	(None, 16, 16, 728)	536536	
['block9_sepconv1_act[0][0]']			

2D)

block9_sepconv1_bn (BatchNorma (None, 16, 16, 728) 2912
['block9_sepconv1[0][0]']
lization)

block9_sepconv2_act (Activatio (None, 16, 16, 728) 0
['block9_sepconv1_bn[0][0]']
n)

block9_sepconv2 (SeparableConv (None, 16, 16, 728) 536536
['block9_sepconv2_act[0][0]']
2D)

block9_sepconv2_bn (BatchNorma (None, 16, 16, 728) 2912
['block9_sepconv2[0][0]']
lization)

block9_sepconv3_act (Activatio (None, 16, 16, 728) 0
['block9_sepconv2_bn[0][0]']
n)

block9_sepconv3 (SeparableConv (None, 16, 16, 728) 536536
['block9_sepconv3_act[0][0]']
2D)

block9_sepconv3_bn (BatchNorma (None, 16, 16, 728) 2912
['block9_sepconv3[0][0]']
lization)

add_7 (Add) (None, 16, 16, 728) 0
['block9_sepconv3_bn[0][0]',
'add_6[0][0]']

block10_sepconv1_act (Activati (None, 16, 16, 728) 0 ['add_7[0][0]']
on)

block10_sepconv1 (SeparableCon (None, 16, 16, 728) 536536
['block10_sepconv1_act[0][0]']
v2D)

block10_sepconv1_bn (BatchNorm (None, 16, 16, 728) 2912
['block10_sepconv1[0][0]']
alization)

block10_sepconv2_act (Activati (None, 16, 16, 728) 0
['block10_sepconv1_bn[0][0]']
on)

```

block10_sepconv2 (SeparableCon (None, 16, 16, 728) 536536
['block10_sepconv2_act[0][0]']
v2D)

block10_sepconv2_bn (BatchNorm (None, 16, 16, 728) 2912
['block10_sepconv2[0][0]']
alization)

block10_sepconv3_act (Activati (None, 16, 16, 728) 0
['block10_sepconv2_bn[0][0]']
on)

block10_sepconv3 (SeparableCon (None, 16, 16, 728) 536536
['block10_sepconv3_act[0][0]']
v2D)

block10_sepconv3_bn (BatchNorm (None, 16, 16, 728) 2912
['block10_sepconv3[0][0]']
alization)

add_8 (Add) (None, 16, 16, 728) 0
['block10_sepconv3_bn[0][0]',
'add_7[0][0]']

block11_sepconv1_act (Activati (None, 16, 16, 728) 0 ['add_8[0][0]']
on)

block11_sepconv1 (SeparableCon (None, 16, 16, 728) 536536
['block11_sepconv1_act[0][0]']
v2D)

block11_sepconv1_bn (BatchNorm (None, 16, 16, 728) 2912
['block11_sepconv1[0][0]']
alization)

block11_sepconv2_act (Activati (None, 16, 16, 728) 0
['block11_sepconv1_bn[0][0]']
on)

block11_sepconv2 (SeparableCon (None, 16, 16, 728) 536536
['block11_sepconv2_act[0][0]']
v2D)

block11_sepconv2_bn (BatchNorm (None, 16, 16, 728) 2912
['block11_sepconv2[0][0]']
alization)

```

```

block11_sepconv3_act (Activation) (None, 16, 16, 728) 0
['block11_sepconv2_bn[0][0]']
on)

block11_sepconv3 (SeparableConv2D) (None, 16, 16, 728) 536536
['block11_sepconv3_act[0][0]']
v2D)

block11_sepconv3_bn (BatchNormalization) (None, 16, 16, 728) 2912
['block11_sepconv3[0][0]']
alization)

add_9 (Add) (None, 16, 16, 728) 0
['block11_sepconv3_bn[0][0]',
'add_8[0][0]']

block12_sepconv1_act (Activation) (None, 16, 16, 728) 0 ['add_9[0][0]']
on)

block12_sepconv1 (SeparableConv2D) (None, 16, 16, 728) 536536
['block12_sepconv1_act[0][0]']
v2D)

block12_sepconv1_bn (BatchNormalization) (None, 16, 16, 728) 2912
['block12_sepconv1[0][0]']
alization)

block12_sepconv2_act (Activation) (None, 16, 16, 728) 0
['block12_sepconv1_bn[0][0]']
on)

block12_sepconv2 (SeparableConv2D) (None, 16, 16, 728) 536536
['block12_sepconv2_act[0][0]']
v2D)

block12_sepconv2_bn (BatchNormalization) (None, 16, 16, 728) 2912
['block12_sepconv2[0][0]']
alization)

block12_sepconv3_act (Activation) (None, 16, 16, 728) 0
['block12_sepconv2_bn[0][0]']
on)

block12_sepconv3 (SeparableConv2D) (None, 16, 16, 728) 536536
['block12_sepconv3_act[0][0]']
v2D)

block12_sepconv3_bn (BatchNormalization) (None, 16, 16, 728) 2912

```



```

['block12_sepconv3[0][0]']
alization)

add_10 (Add) (None, 16, 16, 728) 0
['block12_sepconv3_bn[0][0]',
'add_9[0][0]']

block13_sepconv1_act (Activation) (None, 16, 16, 728) 0
['add_10[0][0]']
on)

block13_sepconv1 (SeparableConv2D) (None, 16, 16, 728) 536536
['block13_sepconv1_act[0][0]']
v2D)

block13_sepconv1_bn (BatchNormalization) (None, 16, 16, 728) 2912
['block13_sepconv1[0][0]']
alization)

block13_sepconv2_act (Activation) (None, 16, 16, 728) 0
['block13_sepconv1_bn[0][0]']
on)

block13_sepconv2 (SeparableConv2D) (None, 16, 16, 1024) 752024
['block13_sepconv2_act[0][0]']
v2D)

block13_sepconv2_bn (BatchNormalization) (None, 16, 16, 1024) 4096
['block13_sepconv2[0][0]']
alization)

conv2d_19 (Conv2D) (None, 8, 8, 1024) 745472
['add_10[0][0]']

block13_pool (MaxPooling2D) (None, 8, 8, 1024) 0
['block13_sepconv2_bn[0][0]']

batch_normalization_19 (BatchNormalization) (None, 8, 8, 1024) 4096
['conv2d_19[0][0]']
ormalization)

add_11 (Add) (None, 8, 8, 1024) 0
['block13_pool[0][0]',
'batch_normalization_19[0][0]']

block14_sepconv1 (SeparableConv2D) (None, 8, 8, 1536) 1582080
['add_11[0][0]']
v2D)

```

```

block14_sepconv1_bn (BatchNorm (None, 8, 8, 1536) 6144
['block14_sepconv1[0][0]']
alization)

block14_sepconv1_act (Activati (None, 8, 8, 1536) 0
['block14_sepconv1_bn[0][0]']
on)

block14_sepconv2 (SeparableCon (None, 8, 8, 2048) 3159552
['block14_sepconv1_act[0][0]']
v2D)

block14_sepconv2_bn (BatchNorm (None, 8, 8, 2048) 8192
['block14_sepconv2[0][0]']
alization)

block14_sepconv2_act (Activati (None, 8, 8, 2048) 0
['block14_sepconv2_bn[0][0]']
on)

global_average_pooling2d_2 (Gl (None, 2048) 0
['block14_sepconv2_act[0][0]']
obalAveragePooling2D)

dense_3 (Dense) (None, 3) 6147
['global_average_pooling2d_2[0][0]']
]']

```

```

=====
=====
Total params: 20,867,627
Trainable params: 6,147
Non-trainable params: 20,861,480
-----
-----

```

3.0.1 Fine-tuning

```

[ ]: model.trainable=True
for layer in model.layers:
    if type(layer) == tf.keras.layers.BatchNormalization:
        layer.trainable=False
    else:
        layer.trainable=True
    assert (type(layer) == tf.keras.layers.BatchNormalization) != layer.
        trainable

```

```
print(sum([type(layer) == tf.keras.layers.BatchNormalization for layer in model.
↳layers]))
model.summary()
```

40

Model: "model"

```
-----
Layer (type)                 Output Shape              Param #   Connected to
=====
input_1 (InputLayer)         [(None, 256, 256, 3      0
)]

block1_conv1 (Conv2D)        (None, 127, 127, 32      864
['input_1[0][0]'])

block1_conv1_bn (BatchNormaliz (None, 127, 127, 32      128
['block1_conv1[0][0]']
ation)

block1_conv1_act (Activation) (None, 127, 127, 32      0
['block1_conv1_bn[0][0]'])

block1_conv2 (Conv2D)        (None, 125, 125, 64      18432
['block1_conv1_act[0][0]'])

block1_conv2_bn (BatchNormaliz (None, 125, 125, 64      256
['block1_conv2[0][0]']
ation)

block1_conv2_act (Activation) (None, 125, 125, 64      0
['block1_conv2_bn[0][0]'])

block2_sepconv1 (SeparableConv (None, 125, 125, 12      8768
['block1_conv2_act[0][0]']
2D)

block2_sepconv1_bn (BatchNorma (None, 125, 125, 12      512
['block2_sepconv1[0][0]']
lization)

block2_sepconv2_act (Activatio (None, 125, 125, 12      0
['block2_sepconv1_bn[0][0]'])
```

```

n)                                     8)

block2_sepconv2 (SeparableConv (None, 125, 125, 12 17536
['block2_sepconv2_act[0][0]']
2D)                                     8)

block2_sepconv2_bn (BatchNorma (None, 125, 125, 12 512
['block2_sepconv2[0][0]']
lization)                               8)

conv2d_16 (Conv2D)                  (None, 63, 63, 128) 8192
['block1_conv2_act[0][0]']

block2_pool (MaxPooling2D)         (None, 63, 63, 128) 0
['block2_sepconv2_bn[0][0]']

batch_normalization_16 (BatchN (None, 63, 63, 128) 512
['conv2d_16[0][0]']
ormalization)

add (Add)                          (None, 63, 63, 128) 0
['block2_pool[0][0]',
'batch_normalization_16[0][0]']

block3_sepconv1_act (Activatio (None, 63, 63, 128) 0      ['add[0][0]']
n)

block3_sepconv1 (SeparableConv (None, 63, 63, 256) 33920
['block3_sepconv1_act[0][0]']
2D)

block3_sepconv1_bn (BatchNorma (None, 63, 63, 256) 1024
['block3_sepconv1[0][0]']
lization)

block3_sepconv2_act (Activatio (None, 63, 63, 256) 0
['block3_sepconv1_bn[0][0]']
n)

block3_sepconv2 (SeparableConv (None, 63, 63, 256) 67840
['block3_sepconv2_act[0][0]']
2D)

block3_sepconv2_bn (BatchNorma (None, 63, 63, 256) 1024
['block3_sepconv2[0][0]']
lization)

conv2d_17 (Conv2D)                (None, 32, 32, 256) 32768      ['add[0][0]']

```

```

block3_pool (MaxPooling2D)      (None, 32, 32, 256)  0
['block3_sepconv2_bn[0][0]']

batch_normalization_17 (BatchN  (None, 32, 32, 256)  1024
['conv2d_17[0][0]']
ormalization)

add_1 (Add)                      (None, 32, 32, 256)  0
['block3_pool[0][0]',
'batch_normalization_17[0][0]']

block4_sepconv1_act (Activatio  (None, 32, 32, 256)  0          ['add_1[0][0]']
n)

block4_sepconv1 (SeparableConv  (None, 32, 32, 728)  188672
['block4_sepconv1_act[0][0]']
2D)

block4_sepconv1_bn (BatchNorma  (None, 32, 32, 728)  2912
['block4_sepconv1[0][0]']
lization)

block4_sepconv2_act (Activatio  (None, 32, 32, 728)  0
['block4_sepconv1_bn[0][0]']
n)

block4_sepconv2 (SeparableConv  (None, 32, 32, 728)  536536
['block4_sepconv2_act[0][0]']
2D)

block4_sepconv2_bn (BatchNorma  (None, 32, 32, 728)  2912
['block4_sepconv2[0][0]']
lization)

conv2d_18 (Conv2D)              (None, 16, 16, 728)  186368          ['add_1[0][0]']

block4_pool (MaxPooling2D)      (None, 16, 16, 728)  0
['block4_sepconv2_bn[0][0]']

batch_normalization_18 (BatchN  (None, 16, 16, 728)  2912
['conv2d_18[0][0]']
ormalization)

add_2 (Add)                      (None, 16, 16, 728)  0
['block4_pool[0][0]',
'batch_normalization_18[0][0]']

```

block5_sepconv1_act (Activation)	(None, 16, 16, 728)	0	['add_2[0][0]']
block5_sepconv1 (SeparableConv2D)	(None, 16, 16, 728)	536536	['block5_sepconv1_act[0][0]']
block5_sepconv1_bn (BatchNormalization)	(None, 16, 16, 728)	2912	['block5_sepconv1[0][0]']
block5_sepconv2_act (Activation)	(None, 16, 16, 728)	0	['block5_sepconv1_bn[0][0]']
block5_sepconv2 (SeparableConv2D)	(None, 16, 16, 728)	536536	['block5_sepconv2_act[0][0]']
block5_sepconv2_bn (BatchNormalization)	(None, 16, 16, 728)	2912	['block5_sepconv2[0][0]']
block5_sepconv3_act (Activation)	(None, 16, 16, 728)	0	['block5_sepconv2_bn[0][0]']
block5_sepconv3 (SeparableConv2D)	(None, 16, 16, 728)	536536	['block5_sepconv3_act[0][0]']
block5_sepconv3_bn (BatchNormalization)	(None, 16, 16, 728)	2912	['block5_sepconv3[0][0]']
add_3 (Add)	(None, 16, 16, 728)	0	['block5_sepconv3_bn[0][0]', 'add_2[0][0]']
block6_sepconv1_act (Activation)	(None, 16, 16, 728)	0	['add_3[0][0]']
block6_sepconv1 (SeparableConv2D)	(None, 16, 16, 728)	536536	['block6_sepconv1_act[0][0]']
block6_sepconv1_bn (BatchNormalization)	(None, 16, 16, 728)	2912	['block6_sepconv1[0][0]']

```

lization)

block6_sepconv2_act (Activation (None, 16, 16, 728) 0
['block6_sepconv1_bn[0][0]']
n)

block6_sepconv2 (SeparableConv (None, 16, 16, 728) 536536
['block6_sepconv2_act[0][0]']
2D)

block6_sepconv2_bn (BatchNormaliza (None, 16, 16, 728) 2912
['block6_sepconv2[0][0]']
lization)

block6_sepconv3_act (Activation (None, 16, 16, 728) 0
['block6_sepconv2_bn[0][0]']
n)

block6_sepconv3 (SeparableConv (None, 16, 16, 728) 536536
['block6_sepconv3_act[0][0]']
2D)

block6_sepconv3_bn (BatchNormaliza (None, 16, 16, 728) 2912
['block6_sepconv3[0][0]']
lization)

add_4 (Add) (None, 16, 16, 728) 0
['block6_sepconv3_bn[0][0]',
'add_3[0][0]']

block7_sepconv1_act (Activation (None, 16, 16, 728) 0 ['add_4[0][0]']
n)

block7_sepconv1 (SeparableConv (None, 16, 16, 728) 536536
['block7_sepconv1_act[0][0]']
2D)

block7_sepconv1_bn (BatchNormaliza (None, 16, 16, 728) 2912
['block7_sepconv1[0][0]']
lization)

block7_sepconv2_act (Activation (None, 16, 16, 728) 0
['block7_sepconv1_bn[0][0]']
n)

block7_sepconv2 (SeparableConv (None, 16, 16, 728) 536536
['block7_sepconv2_act[0][0]']
2D)

```

```

block7_sepconv2_bn (BatchNorma (None, 16, 16, 728) 2912
['block7_sepconv2[0][0]']
lization)

block7_sepconv3_act (Activatio (None, 16, 16, 728) 0
['block7_sepconv2_bn[0][0]']
n)

block7_sepconv3 (SeparableConv (None, 16, 16, 728) 536536
['block7_sepconv3_act[0][0]']
2D)

block7_sepconv3_bn (BatchNorma (None, 16, 16, 728) 2912
['block7_sepconv3[0][0]']
lization)

add_5 (Add) (None, 16, 16, 728) 0
['block7_sepconv3_bn[0][0]',
'add_4[0][0]']

block8_sepconv1_act (Activatio (None, 16, 16, 728) 0 ['add_5[0][0]']
n)

block8_sepconv1 (SeparableConv (None, 16, 16, 728) 536536
['block8_sepconv1_act[0][0]']
2D)

block8_sepconv1_bn (BatchNorma (None, 16, 16, 728) 2912
['block8_sepconv1[0][0]']
lization)

block8_sepconv2_act (Activatio (None, 16, 16, 728) 0
['block8_sepconv1_bn[0][0]']
n)

block8_sepconv2 (SeparableConv (None, 16, 16, 728) 536536
['block8_sepconv2_act[0][0]']
2D)

block8_sepconv2_bn (BatchNorma (None, 16, 16, 728) 2912
['block8_sepconv2[0][0]']
lization)

block8_sepconv3_act (Activatio (None, 16, 16, 728) 0
['block8_sepconv2_bn[0][0]']
n)

```



```

block8_sepconv3 (SeparableConv (None, 16, 16, 728) 536536
['block8_sepconv3_act[0][0]']
2D)

block8_sepconv3_bn (BatchNorma (None, 16, 16, 728) 2912
['block8_sepconv3[0][0]']
lization)

add_6 (Add) (None, 16, 16, 728) 0
['block8_sepconv3_bn[0][0]',
'add_5[0][0]']

block9_sepconv1_act (Activatio (None, 16, 16, 728) 0 ['add_6[0][0]']
n)

block9_sepconv1 (SeparableConv (None, 16, 16, 728) 536536
['block9_sepconv1_act[0][0]']
2D)

block9_sepconv1_bn (BatchNorma (None, 16, 16, 728) 2912
['block9_sepconv1[0][0]']
lization)

block9_sepconv2_act (Activatio (None, 16, 16, 728) 0
['block9_sepconv1_bn[0][0]']
n)

block9_sepconv2 (SeparableConv (None, 16, 16, 728) 536536
['block9_sepconv2_act[0][0]']
2D)

block9_sepconv2_bn (BatchNorma (None, 16, 16, 728) 2912
['block9_sepconv2[0][0]']
lization)

block9_sepconv3_act (Activatio (None, 16, 16, 728) 0
['block9_sepconv2_bn[0][0]']
n)

block9_sepconv3 (SeparableConv (None, 16, 16, 728) 536536
['block9_sepconv3_act[0][0]']
2D)

block9_sepconv3_bn (BatchNorma (None, 16, 16, 728) 2912
['block9_sepconv3[0][0]']
lization)

add_7 (Add) (None, 16, 16, 728) 0

```

['block9_sepconv3_bn[0][0]',					'add_6[0][0]'
block10_sepconv1_act (Activati on)	(None, 16, 16, 728)	0			['add_7[0][0]']
block10_sepconv1 (SeparableCon ['block10_sepconv1_act[0][0]' v2D)	(None, 16, 16, 728)	536536			
block10_sepconv1_bn (BatchNorm ['block10_sepconv1[0][0]' alization)	(None, 16, 16, 728)	2912			
block10_sepconv2_act (Activati ['block10_sepconv1_bn[0][0]' on)	(None, 16, 16, 728)	0			
block10_sepconv2 (SeparableCon ['block10_sepconv2_act[0][0]' v2D)	(None, 16, 16, 728)	536536			
block10_sepconv2_bn (BatchNorm ['block10_sepconv2[0][0]' alization)	(None, 16, 16, 728)	2912			
block10_sepconv3_act (Activati ['block10_sepconv2_bn[0][0]' on)	(None, 16, 16, 728)	0			
block10_sepconv3 (SeparableCon ['block10_sepconv3_act[0][0]' v2D)	(None, 16, 16, 728)	536536			
block10_sepconv3_bn (BatchNorm ['block10_sepconv3[0][0]' alization)	(None, 16, 16, 728)	2912			
add_8 (Add) ['block10_sepconv3_bn[0][0]',	(None, 16, 16, 728)	0			'add_7[0][0]'
block11_sepconv1_act (Activati on)	(None, 16, 16, 728)	0			['add_8[0][0]']
block11_sepconv1 (SeparableCon ['block11_sepconv1_act[0][0]' v2D)	(None, 16, 16, 728)	536536			

```

block11_sepconv1_bn (BatchNorm (None, 16, 16, 728) 2912
['block11_sepconv1[0][0]']
alization)

block11_sepconv2_act (Activati (None, 16, 16, 728) 0
['block11_sepconv1_bn[0][0]']
on)

block11_sepconv2 (SeparableCon (None, 16, 16, 728) 536536
['block11_sepconv2_act[0][0]']
v2D)

block11_sepconv2_bn (BatchNorm (None, 16, 16, 728) 2912
['block11_sepconv2[0][0]']
alization)

block11_sepconv3_act (Activati (None, 16, 16, 728) 0
['block11_sepconv2_bn[0][0]']
on)

block11_sepconv3 (SeparableCon (None, 16, 16, 728) 536536
['block11_sepconv3_act[0][0]']
v2D)

block11_sepconv3_bn (BatchNorm (None, 16, 16, 728) 2912
['block11_sepconv3[0][0]']
alization)

add_9 (Add) (None, 16, 16, 728) 0
['block11_sepconv3_bn[0][0]',
'add_8[0][0]']

block12_sepconv1_act (Activati (None, 16, 16, 728) 0 ['add_9[0][0]']
on)

block12_sepconv1 (SeparableCon (None, 16, 16, 728) 536536
['block12_sepconv1_act[0][0]']
v2D)

block12_sepconv1_bn (BatchNorm (None, 16, 16, 728) 2912
['block12_sepconv1[0][0]']
alization)

block12_sepconv2_act (Activati (None, 16, 16, 728) 0
['block12_sepconv1_bn[0][0]']
on)

```

```

block12_sepconv2 (SeparableCon (None, 16, 16, 728) 536536
['block12_sepconv2_act[0][0]']
v2D)

block12_sepconv2_bn (BatchNorm (None, 16, 16, 728) 2912
['block12_sepconv2[0][0]']
alization)

block12_sepconv3_act (Activati (None, 16, 16, 728) 0
['block12_sepconv2_bn[0][0]']
on)

block12_sepconv3 (SeparableCon (None, 16, 16, 728) 536536
['block12_sepconv3_act[0][0]']
v2D)

block12_sepconv3_bn (BatchNorm (None, 16, 16, 728) 2912
['block12_sepconv3[0][0]']
alization)

add_10 (Add) (None, 16, 16, 728) 0
['block12_sepconv3_bn[0][0]',
'add_9[0][0]']

block13_sepconv1_act (Activati (None, 16, 16, 728) 0
['add_10[0][0]']
on)

block13_sepconv1 (SeparableCon (None, 16, 16, 728) 536536
['block13_sepconv1_act[0][0]']
v2D)

block13_sepconv1_bn (BatchNorm (None, 16, 16, 728) 2912
['block13_sepconv1[0][0]']
alization)

block13_sepconv2_act (Activati (None, 16, 16, 728) 0
['block13_sepconv1_bn[0][0]']
on)

block13_sepconv2 (SeparableCon (None, 16, 16, 1024 752024
['block13_sepconv2_act[0][0]']
v2D) )

block13_sepconv2_bn (BatchNorm (None, 16, 16, 1024 4096
['block13_sepconv2[0][0]']
alization) )

```

conv2d_19 (Conv2D)	(None, 8, 8, 1024)	745472
['add_10[0][0]']		
block13_pool (MaxPooling2D)	(None, 8, 8, 1024)	0
['block13_sepconv2_bn[0][0]']		
batch_normalization_19 (Batch Normalization)	(None, 8, 8, 1024)	4096
['conv2d_19[0][0]']		
add_11 (Add)	(None, 8, 8, 1024)	0
['block13_pool[0][0]', 'batch_normalization_19[0][0]']		
block14_sepconv1 (SeparableConv2D)	(None, 8, 8, 1536)	1582080
['add_11[0][0]']		
block14_sepconv1_bn (Batch Normalization)	(None, 8, 8, 1536)	6144
['block14_sepconv1[0][0]']		
block14_sepconv1_act (Activation)	(None, 8, 8, 1536)	0
['block14_sepconv1_bn[0][0]']		
block14_sepconv2 (SeparableConv2D)	(None, 8, 8, 2048)	3159552
['block14_sepconv1_act[0][0]']		
block14_sepconv2_bn (Batch Normalization)	(None, 8, 8, 2048)	8192
['block14_sepconv2[0][0]']		
block14_sepconv2_act (Activation)	(None, 8, 8, 2048)	0
['block14_sepconv2_bn[0][0]']		
global_average_pooling2d_2 (GlobalAveragePooling2D)	(None, 2048)	0
['block14_sepconv2_act[0][0]']		
dense_3 (Dense)	(None, 3)	6147
['global_average_pooling2d_2[0][0]']		

=====

=====

Total params: 20,867,627
Trainable params: 20,758,571
Non-trainable params: 109,056

Rozmroziłem wszystkie warstwy, poza `tf.keras.layers.BatchNormalization`.
Ustawienie `batchnormalization.training` na `False`, właśnie wprowadza normalizację w taki pośredni stan jak opisany w treści zadania.

```
[ ]: model.compile(  
    optimizer=tf.keras.optimizers.SGD(learning_rate=0.0001, momentum=0.9),  
    loss=tf.keras.losses.CategoricalCrossentropy(),  
    metrics=['accuracy']  
)
```

```
[ ]: history = model.fit(  
    x=train_ds,  
    batch_size=64,  
    epochs=30,  
    verbose=1,  
    validation_data=test_ds  
)
```

Epoch 1/30

2022-11-04 00:41:10.965528: I
tensorflow/core/grappler/optimizers/custom_graph_optimizer_registry.cc:114]
Plugin optimizer for device_type GPU is enabled.

12/12 [=====] - ETA: 0s - loss: 0.0765 - accuracy:
0.9908

2022-11-04 00:41:32.677847: I
tensorflow/core/grappler/optimizers/custom_graph_optimizer_registry.cc:114]
Plugin optimizer for device_type GPU is enabled.

12/12 [=====] - 25s 2s/step - loss: 0.0765 - accuracy:
0.9908 - val_loss: 0.0829 - val_accuracy: 0.9894

Epoch 2/30

12/12 [=====] - 22s 2s/step - loss: 0.0721 - accuracy:
0.9908 - val_loss: 0.0778 - val_accuracy: 0.9894

Epoch 3/30

12/12 [=====] - 21s 2s/step - loss: 0.0667 - accuracy:
0.9921 - val_loss: 0.0728 - val_accuracy: 0.9894

Epoch 4/30

12/12 [=====] - 21s 2s/step - loss: 0.0617 - accuracy:
0.9934 - val_loss: 0.0689 - val_accuracy: 0.9894

Epoch 5/30

12/12 [=====] - 21s 2s/step - loss: 0.0573 - accuracy:
0.9934 - val_loss: 0.0654 - val_accuracy: 0.9894

Epoch 6/30
12/12 [=====] - 21s 2s/step - loss: 0.0535 - accuracy:
0.9934 - val_loss: 0.0623 - val_accuracy: 0.9894

Epoch 7/30
12/12 [=====] - 21s 2s/step - loss: 0.0500 - accuracy:
0.9947 - val_loss: 0.0597 - val_accuracy: 0.9894

Epoch 8/30
12/12 [=====] - 21s 2s/step - loss: 0.0470 - accuracy:
0.9947 - val_loss: 0.0574 - val_accuracy: 0.9894

Epoch 9/30
12/12 [=====] - 21s 2s/step - loss: 0.0443 - accuracy:
0.9947 - val_loss: 0.0554 - val_accuracy: 0.9894

Epoch 10/30
12/12 [=====] - 21s 2s/step - loss: 0.0418 - accuracy:
0.9960 - val_loss: 0.0535 - val_accuracy: 0.9894

Epoch 11/30
12/12 [=====] - 21s 2s/step - loss: 0.0395 - accuracy:
0.9960 - val_loss: 0.0520 - val_accuracy: 0.9894

Epoch 12/30
12/12 [=====] - 21s 2s/step - loss: 0.0373 - accuracy:
0.9974 - val_loss: 0.0505 - val_accuracy: 0.9894

Epoch 13/30
12/12 [=====] - 21s 2s/step - loss: 0.0355 - accuracy:
0.9974 - val_loss: 0.0492 - val_accuracy: 0.9894

Epoch 14/30
12/12 [=====] - 21s 2s/step - loss: 0.0336 - accuracy:
0.9987 - val_loss: 0.0479 - val_accuracy: 0.9894

Epoch 15/30
12/12 [=====] - 21s 2s/step - loss: 0.0320 - accuracy:
0.9987 - val_loss: 0.0466 - val_accuracy: 0.9894

Epoch 16/30
12/12 [=====] - 21s 2s/step - loss: 0.0307 - accuracy:
0.9987 - val_loss: 0.0457 - val_accuracy: 0.9894

Epoch 17/30
12/12 [=====] - 21s 2s/step - loss: 0.0290 - accuracy:
0.9987 - val_loss: 0.0445 - val_accuracy: 0.9894

Epoch 18/30
12/12 [=====] - 21s 2s/step - loss: 0.0277 - accuracy:
0.9987 - val_loss: 0.0435 - val_accuracy: 0.9894

Epoch 19/30
12/12 [=====] - 21s 2s/step - loss: 0.0264 - accuracy:
0.9987 - val_loss: 0.0427 - val_accuracy: 0.9894

Epoch 20/30
12/12 [=====] - 21s 2s/step - loss: 0.0253 - accuracy:
0.9987 - val_loss: 0.0418 - val_accuracy: 0.9894

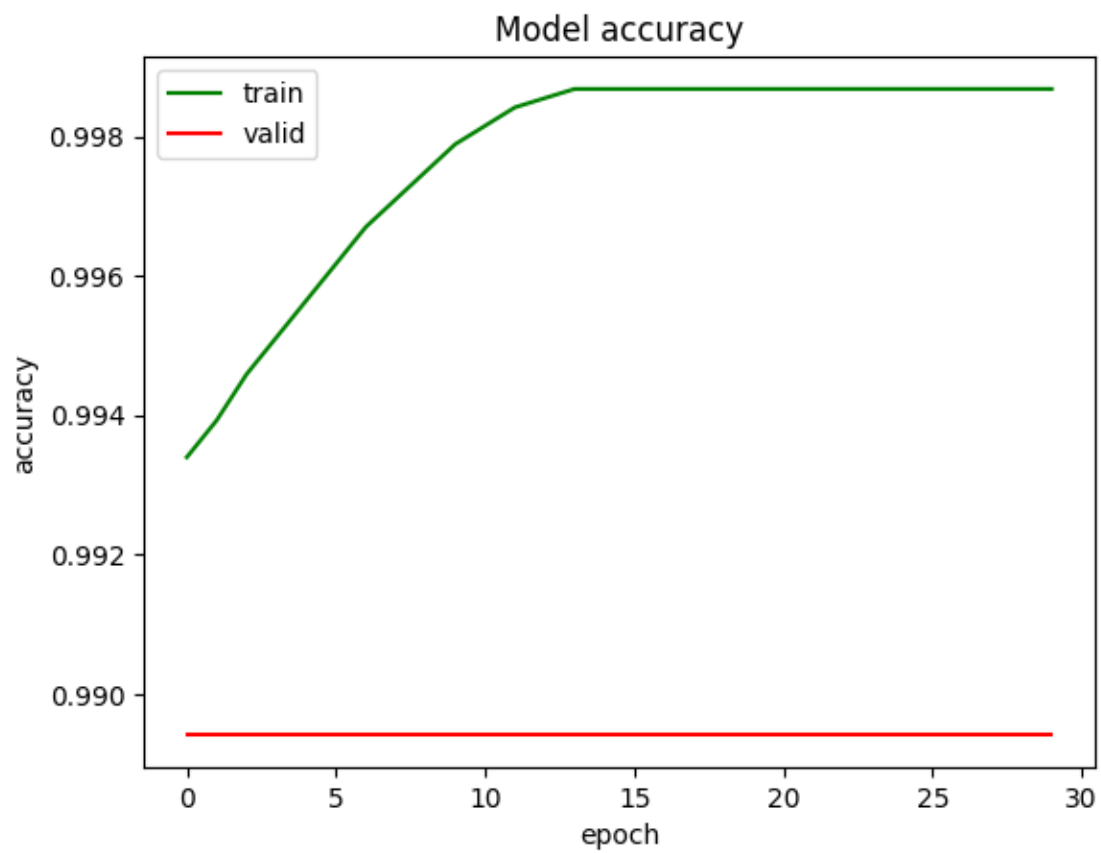
Epoch 21/30
12/12 [=====] - 21s 2s/step - loss: 0.0242 - accuracy:
0.9987 - val_loss: 0.0412 - val_accuracy: 0.9894

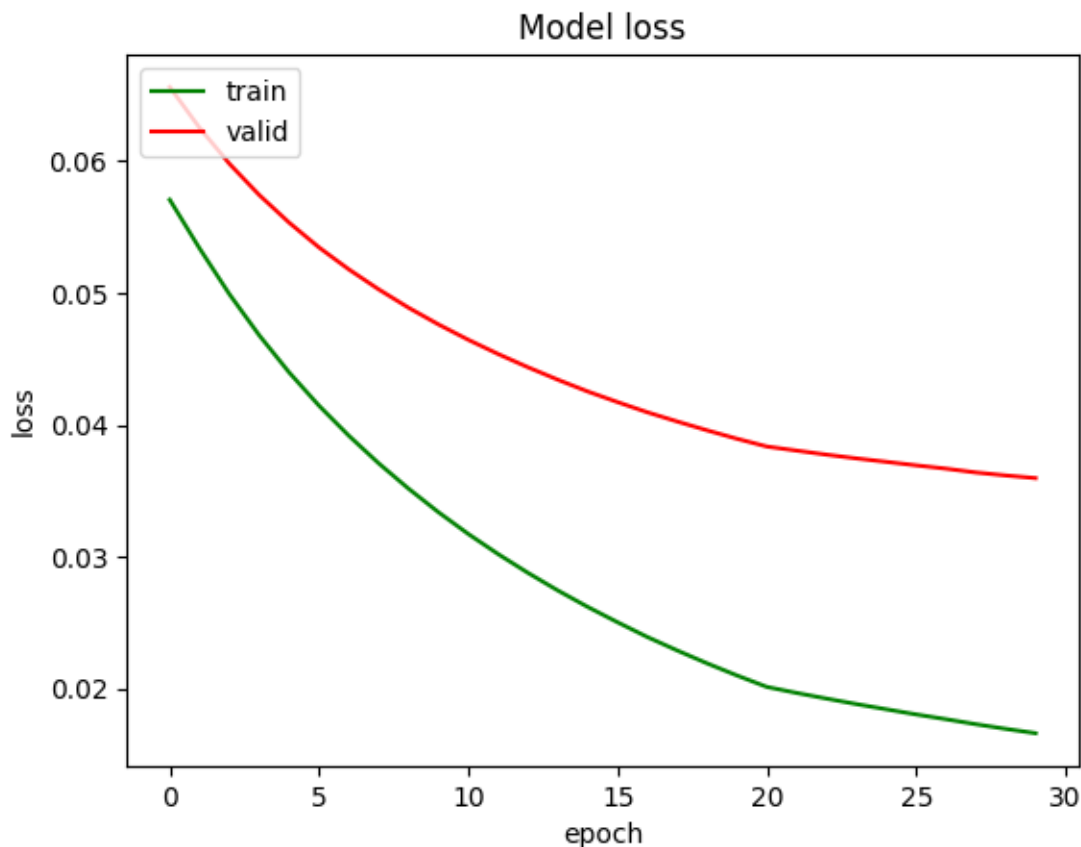
```

Epoch 22/30
12/12 [=====] - 21s 2s/step - loss: 0.0231 - accuracy:
0.9987 - val_loss: 0.0405 - val_accuracy: 0.9894
Epoch 23/30
12/12 [=====] - 21s 2s/step - loss: 0.0221 - accuracy:
0.9987 - val_loss: 0.0397 - val_accuracy: 0.9894
Epoch 24/30
12/12 [=====] - 21s 2s/step - loss: 0.0212 - accuracy:
0.9987 - val_loss: 0.0390 - val_accuracy: 0.9894
Epoch 25/30
12/12 [=====] - 21s 2s/step - loss: 0.0204 - accuracy:
0.9987 - val_loss: 0.0385 - val_accuracy: 0.9894
Epoch 26/30
12/12 [=====] - 21s 2s/step - loss: 0.0195 - accuracy:
0.9987 - val_loss: 0.0379 - val_accuracy: 0.9894
Epoch 27/30
12/12 [=====] - 21s 2s/step - loss: 0.0188 - accuracy:
0.9987 - val_loss: 0.0375 - val_accuracy: 0.9894
Epoch 28/30
12/12 [=====] - 21s 2s/step - loss: 0.0180 - accuracy:
0.9987 - val_loss: 0.0368 - val_accuracy: 0.9894
Epoch 29/30
12/12 [=====] - 21s 2s/step - loss: 0.0173 - accuracy:
0.9987 - val_loss: 0.0364 - val_accuracy: 0.9894
Epoch 30/30
12/12 [=====] - 21s 2s/step - loss: 0.0166 - accuracy:
0.9987 - val_loss: 0.0360 - val_accuracy: 0.9894

```

```
[ ]: print_history(history.history)
```



```
[ ]: model.save_weights('./checkpoints/imagenet_finetuning')
      model.save('./saved_model/imagenet_finetuning')
```

WARNING:absl:Found untraced functions such as _jit_compiled_convolution_op, _jit_compiled_convolution_op, _jit_compiled_convolution_op, _jit_compiled_convolution_op while saving (showing 5 of 40). These functions will not be directly callable after loading.

INFO:tensorflow:Assets written to: ./saved_model/imagenet_finetuning/assets

INFO:tensorflow:Assets written to: ./saved_model/imagenet_finetuning/assets

Na tak przygotowanym modelu, udało się osiągnąć na prawdę dobry wynik.

Jak widać fine tuning bardzo dobrze się sprawdza w takich sytuacjach.

Otrzymujemy bardzo dobre accuracy i stale zmniejszamy loss.

Co ciekawe, accuracy na zbiorze walidacyjnym się nie zmienia, ale myślę, że wynika to z małego rozmiaru zbioru walidacyjnego oraz jakiegoś bardzo trudnego zdjęcia.

4 Decyzje?

```
[ ]: def get_5_each():
    res = [], [], []
    for img, label in test_ds:
        print(tf.math.argmax(label, axis=1))
        res[0] = img[tf.math.argmax(label, axis=1)==0][:5]
        res[1] = img[tf.math.argmax(label, axis=1)==1][:5]
        res[2] = img[tf.math.argmax(label, axis=1)==2][:5]
    return res

fives = get_5_each()

tf.Tensor(
[1 0 0 1 1 2 0 1 1 0 1 0 0 1 1 0 2 1 0 2 2 0 0 0 2 1 2 0 2 0 1 1 1 0 2 2 2
 1 1 2 0 1 1 2 0 0 0 2 1 0 1 2 0 1 1 1 1 0 1 2 0 2 2], shape=(64,),
dtype=int64)
```

```
[ ]: print(len(fives), len(fives[0]), len(fives[1]), len(fives[2]), tf.
    ↪expand_dims(fives[0][1].shape, 0))
```

```
3 5 5 5 tf.Tensor([[256 256   3]], shape=(1, 3), dtype=int32)
```

Niestety mam straszne trudności zarządzać tym datasetem - stąd funkcja `get_5_each`

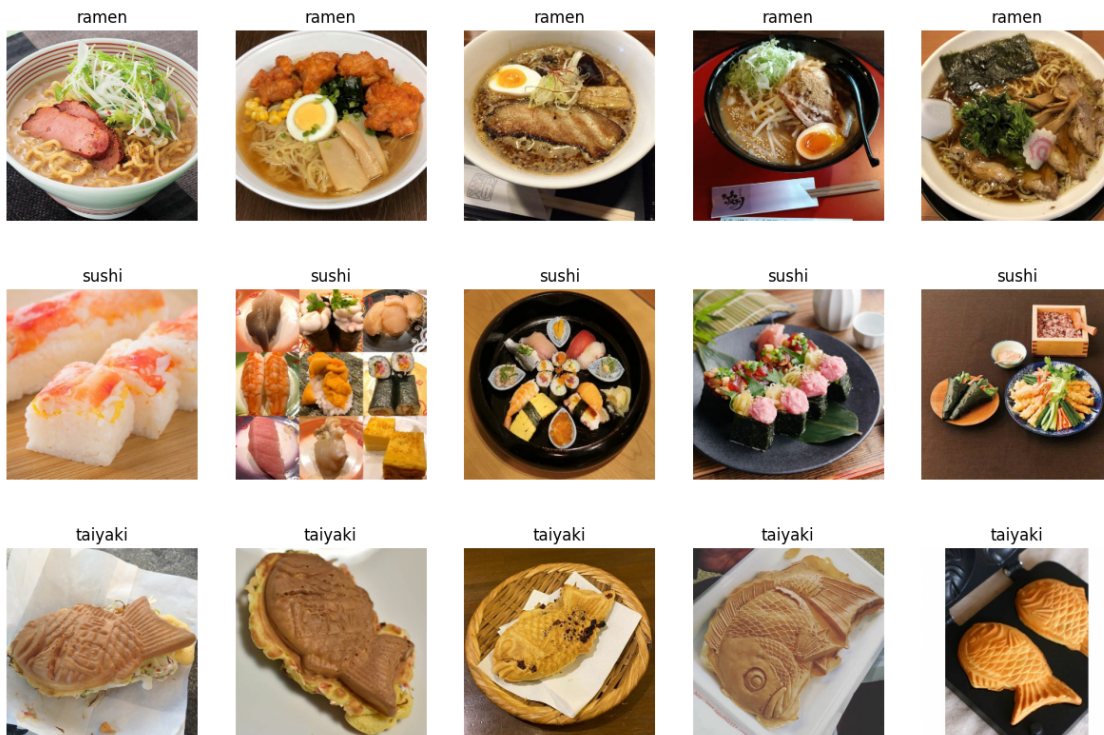
```
[ ]: for ix, cat in enumerate(fives):
    for img in cat:
        assert (tf.math.argmax(model.predict(tf.expand_dims(img,0)),
    ↪axis=1)==ix)
```

```
2022-11-04 01:00:01.988475: I
tensorflow/core/grappler/optimizers/custom_graph_optimizer_registry.cc:114]
Plugin optimizer for device_type GPU is enabled.
```

```
1/1 [=====] - 1s 1s/step
1/1 [=====] - 0s 69ms/step
1/1 [=====] - 0s 25ms/step
1/1 [=====] - 0s 23ms/step
1/1 [=====] - 0s 24ms/step
1/1 [=====] - 0s 24ms/step
1/1 [=====] - 0s 22ms/step
1/1 [=====] - 0s 24ms/step
1/1 [=====] - 0s 24ms/step
1/1 [=====] - 0s 23ms/step
1/1 [=====] - 0s 22ms/step
1/1 [=====] - 0s 23ms/step
1/1 [=====] - 0s 24ms/step
1/1 [=====] - 0s 23ms/step
1/1 [=====] - 0s 22ms/step
```

Upewniłem się, że model poprawnie zgaduje te zdjęcia.

```
[ ]: plt.figure(figsize=(15, 10))
for ix, cat in enumerate(fives):
    for ixx, img in enumerate(cat):
        ax = plt.subplot(3, 5, ixx + 1 + ix*5)
        plt.imshow(img)
        plt.title(class_names[ix])
        plt.axis("off")
        plt.yticks([])
        plt.xticks([])
```



```
[ ]: model.predict(tf.expand_dims(fives[0][0], 0))
```

1/1 [=====] - 0s 26ms/step

```
[ ]: array([[9.9987638e-01, 8.8199653e-05, 3.5363988e-05]], dtype=float32)
```

```
[ ]: model.layers[-5:], model.layers[-3]
```

```
[ ]: ([<keras.layers.convolutional.separable_conv2d.SeparableConv2D at 0x2cab0bdc0>,
      <keras.layers.normalization.batch_normalization.BatchNormalization at
0x2d3adac70>,
      <keras.layers.core.activation.Activation at 0x2cab0b9a0>],
```

```

    <keras.layers.pooling.global_average_pooling2d.GlobalAveragePooling2D at
0x2d2dbfa30>,
    <keras.layers.core.dense.Dense at 0x2d2dbf3a0>],
    <keras.layers.core.activation.Activation at 0x2cab0b9a0>)

```

```

[ ]: tmp_model = tf.keras.models.Model(inputs=model.input,outputs=model.layers[-3].
    ↪output)

```

```

[ ]: tmp_model.output_shape

```

```

[ ]: (None, 8, 8, 2048)

```

```

[ ]: res = tmp_model.predict(tf.expand_dims(fives[0][0], 0))

```

```

1/1 [=====] - 0s 28ms/step

```

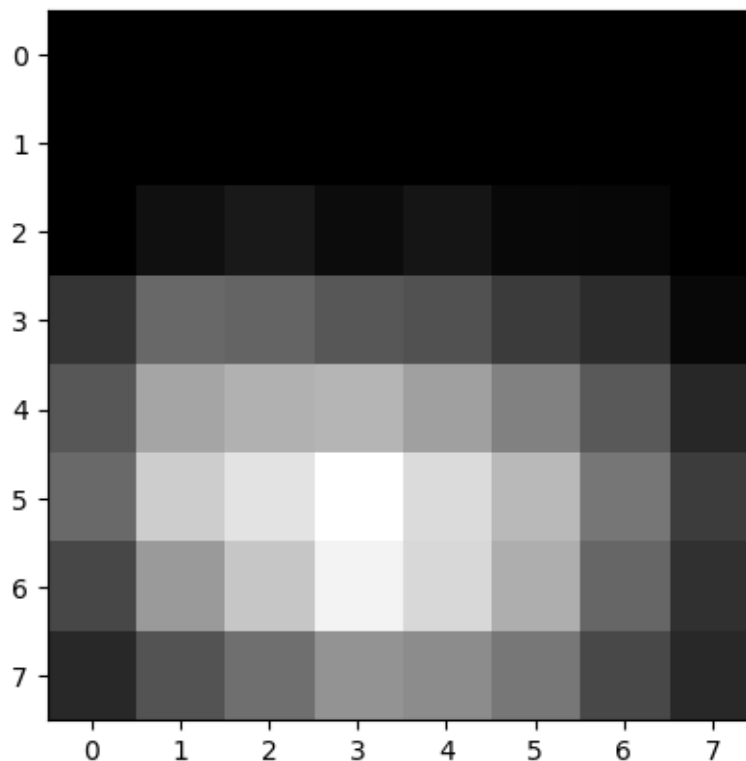
```

[ ]: def print_gray(img):
    plt.imshow(img, cmap='gray')
    plt.show()

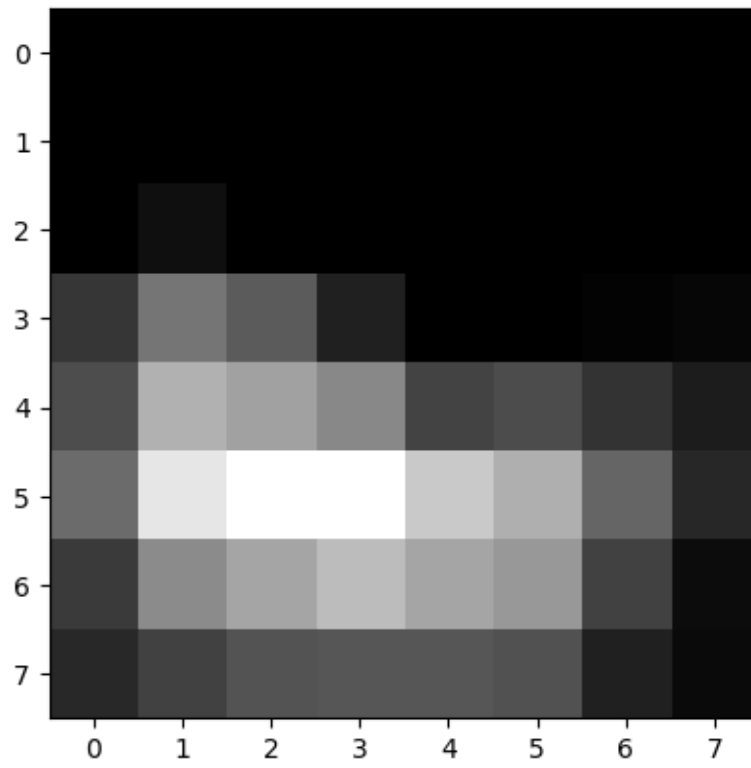
res[0,:,:,:].shape
for channel in np.random.randint(0, 2048, 5):
    print(channel)
    print_gray(res[0,:,:,:channel])

```

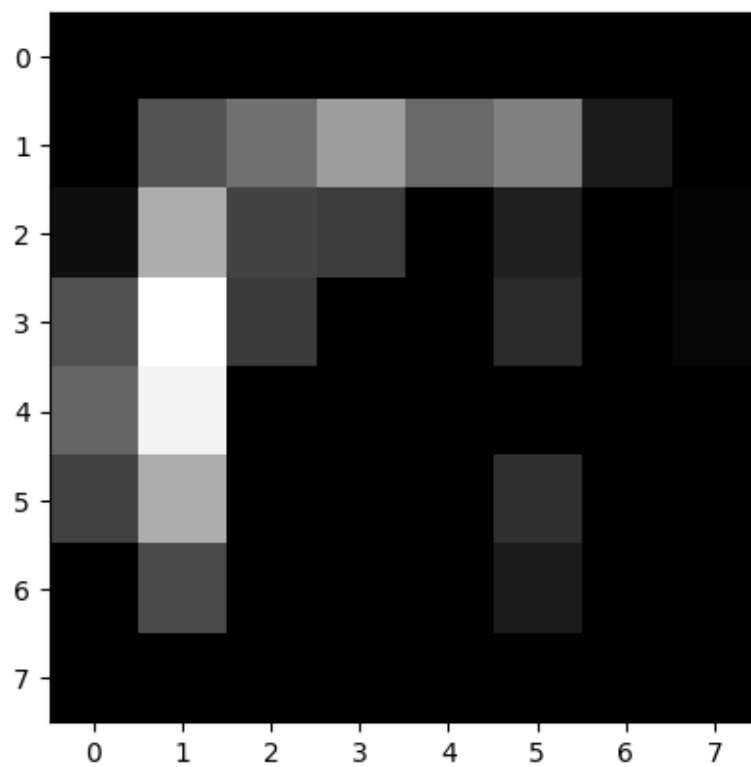
742



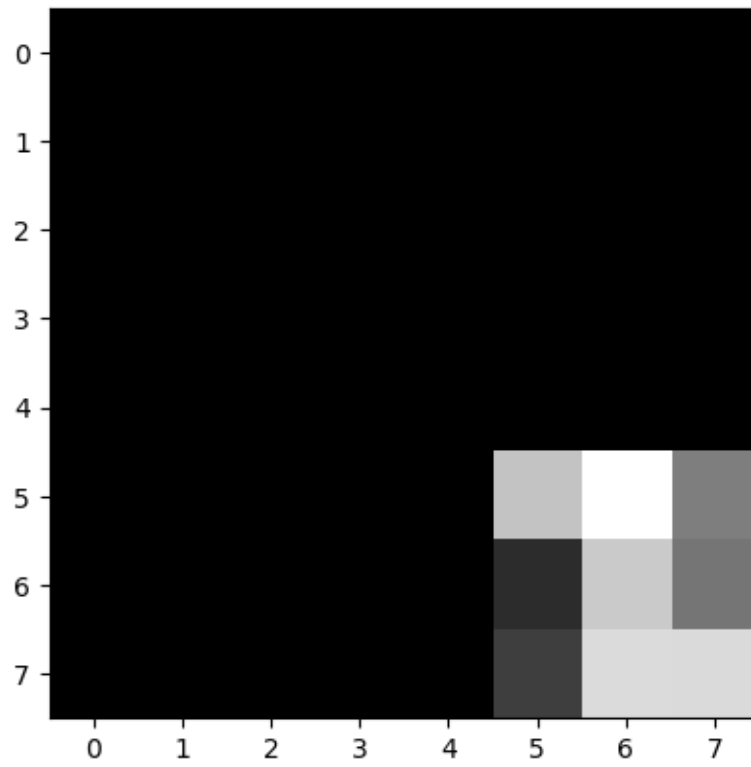
1973



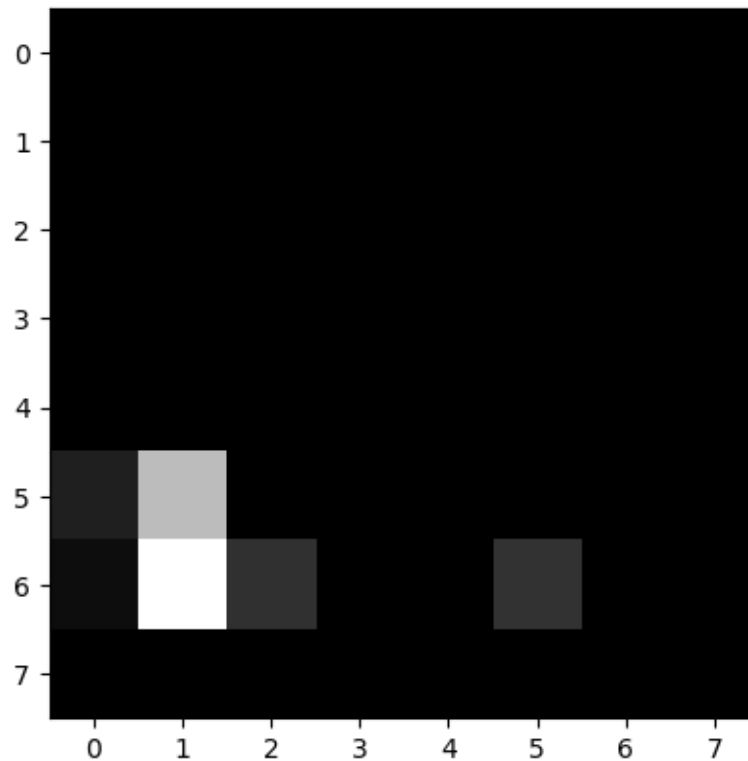
845



763



469



Interesujące, ale ciężko coś na ten temat powiedzieć.

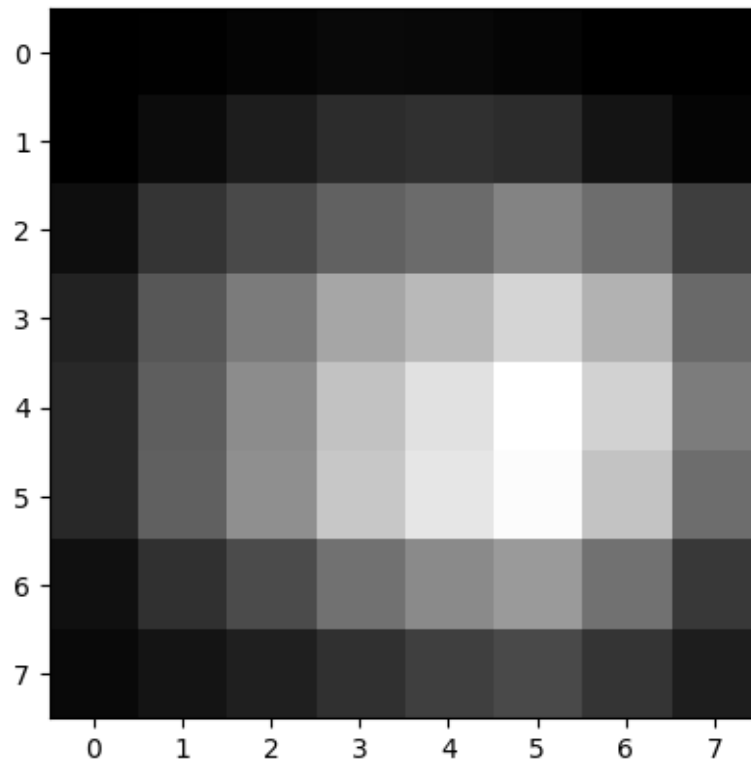
Trochę za dużo kanałów aby pojąć.

Możemy natomiast zrozumieć jakie fragmenty obrazka jak duży miały wpływ na poszczególne kanały wynikowe.

```
[ ]: def get_heat(x, weights, cat):
    return tf.math.reduce_sum(x * weights[:,cat], axis=2)

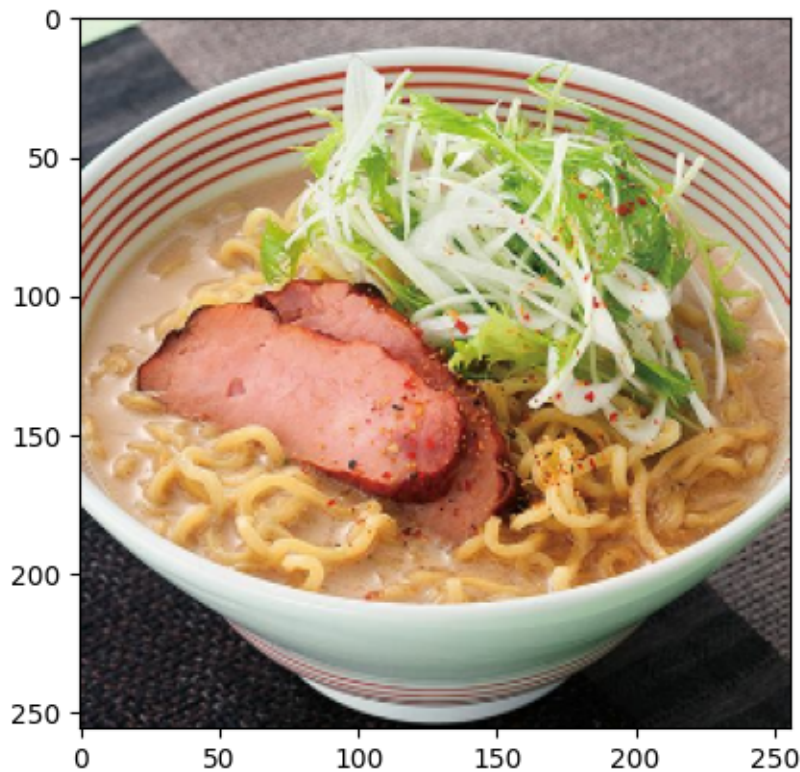
def show_heatmap(x, weights, cat):
    res = get_heat(x, weights, cat)
    print_gray(res)

show_heatmap(res[0], model.layers[-1].get_weights()[0], 0)
```



Do wykonania zadania wykorzystałem operacje na tensorach oraz `tf.math.reduce.sum()`.

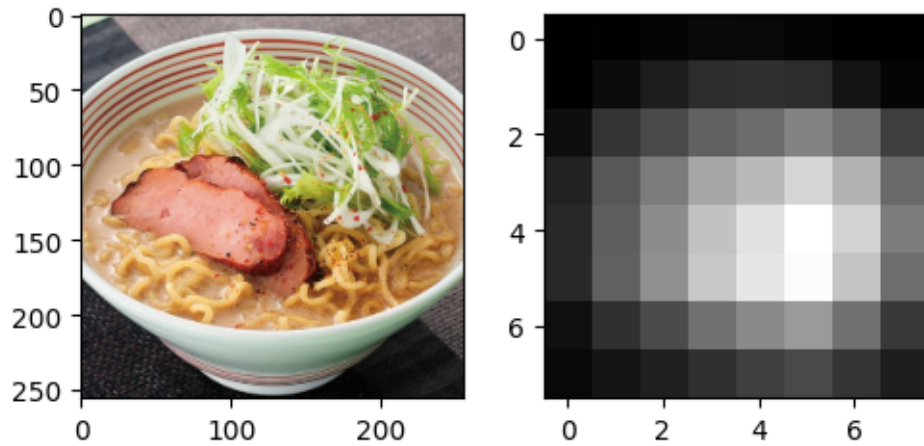
```
[ ]: def print_image(img):  
    plt.imshow(img)  
    plt.show()  
  
print_image(fives[0][0])
```



Wygląda na to, że nasza heatmapa bardzo słusznie skupiła się na misce z ramenem, w szczególności zawartości ramenu.

Ma to zaskakująco dużo sensu.

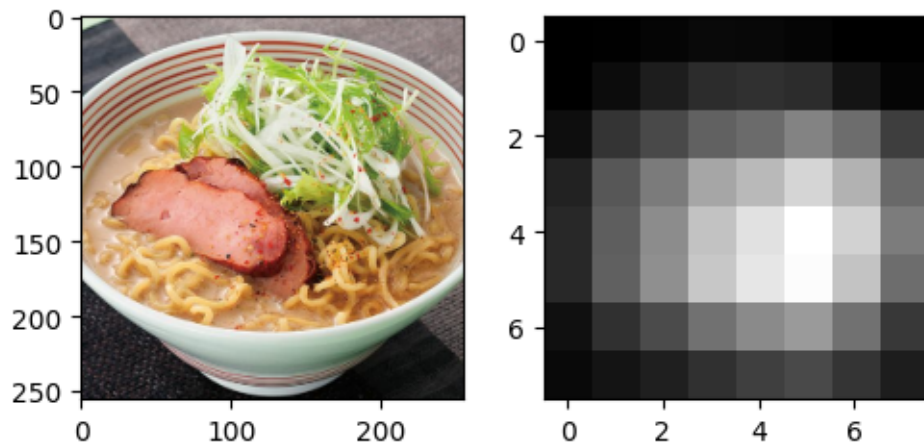
```
[ ]: def print_both(img, heat):  
    plt.figure(figsize=(15, 15))  
    plt.subplot(5,5,1)  
    plt.imshow(img)  
    plt.subplot(5,5,2)  
    plt.imshow(heat, cmap='gray')  
    plt.show()  
  
print_both(fives[0][0], get_heat(res[0], model.layers[-1].get_weights()[0]))
```



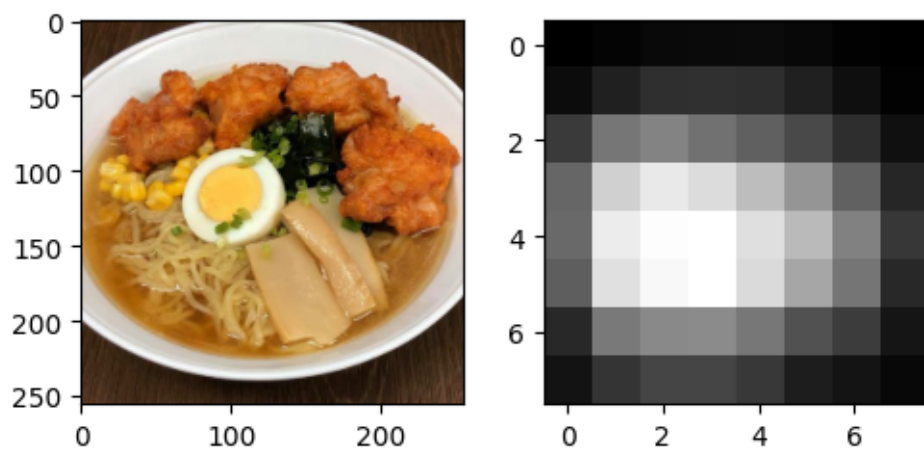
```
[ ]: def compare_heats(model, tmp_model, img, cat):
    print_both(img,
               get_heat(tmp_model.predict(tf.expand_dims(img, 0))[0], model.layers[-1].
               ↪get_weights()[0], cat))

    for cat, five in enumerate(fives):
        for img in five:
            compare_heats(model=model, tmp_model=tmp_model, img=img, cat=cat)
```

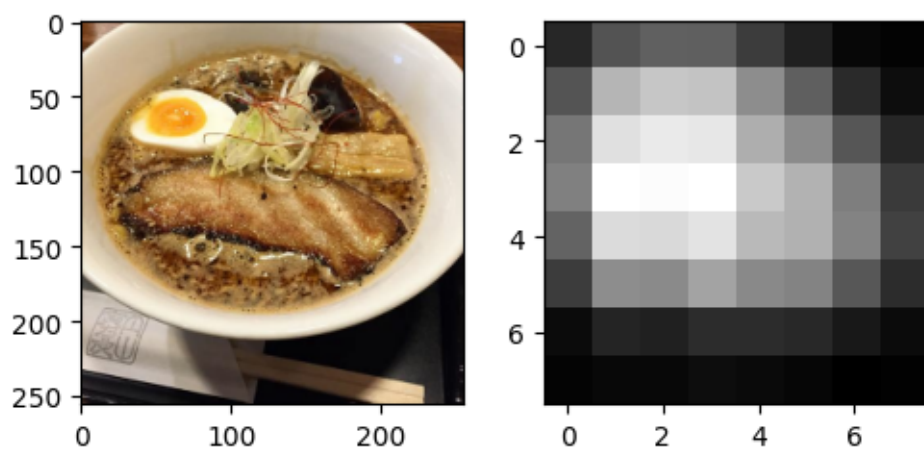
1/1 [=====] - 0s 63ms/step



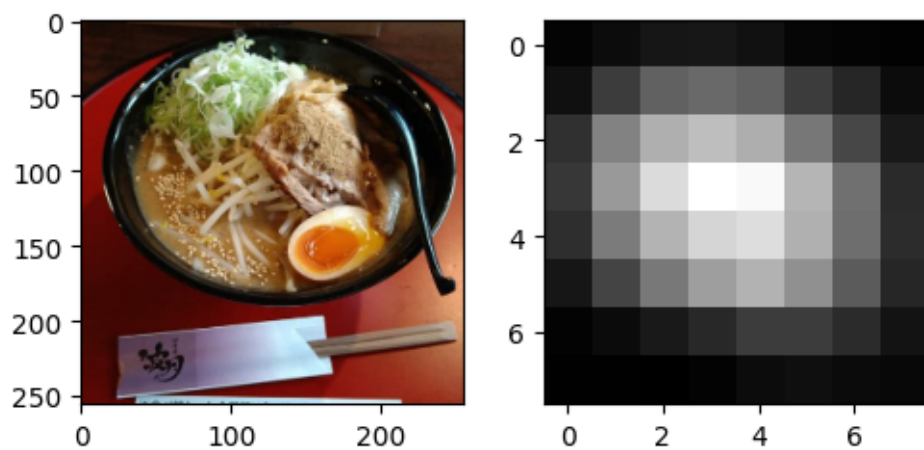
1/1 [=====] - 0s 99ms/step



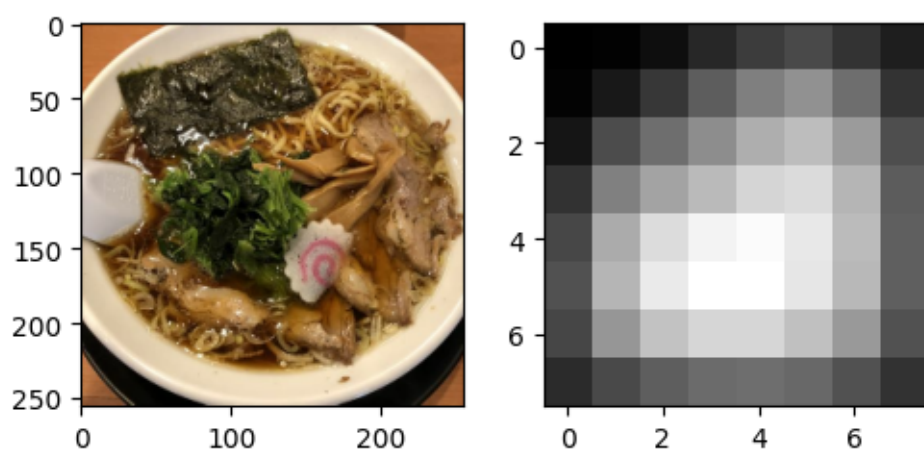
1/1 [=====] - 0s 24ms/step



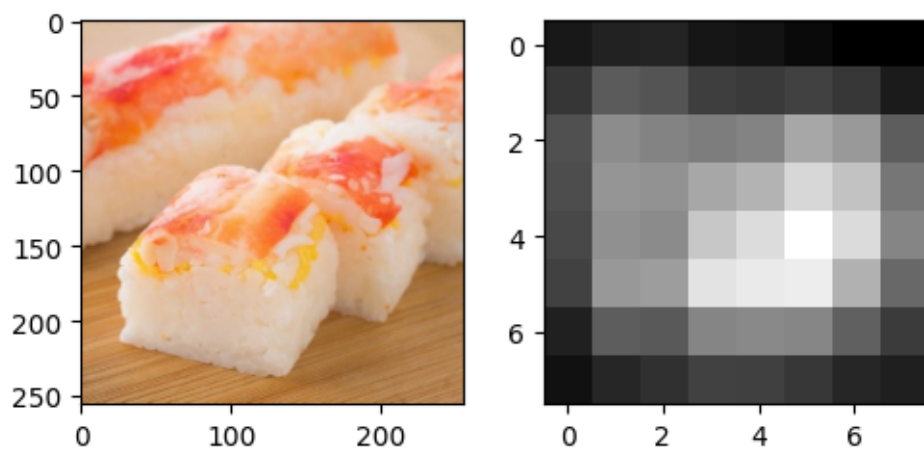
1/1 [=====] - 0s 23ms/step



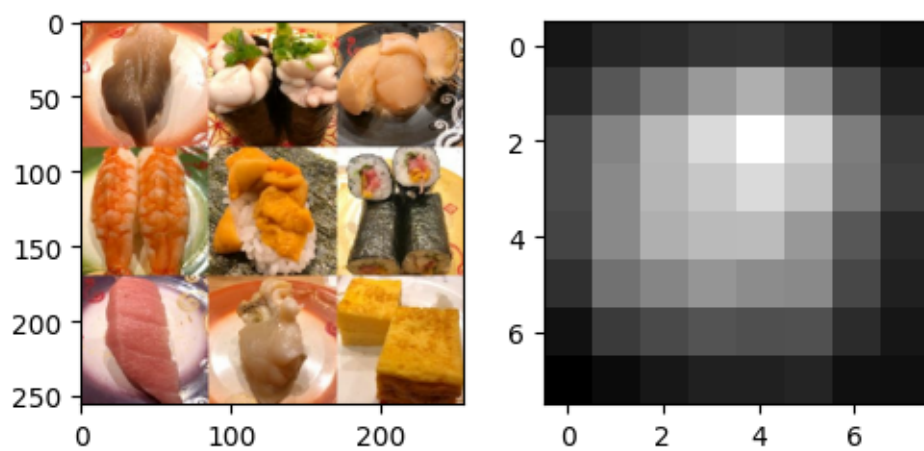
1/1 [=====] - 0s 23ms/step



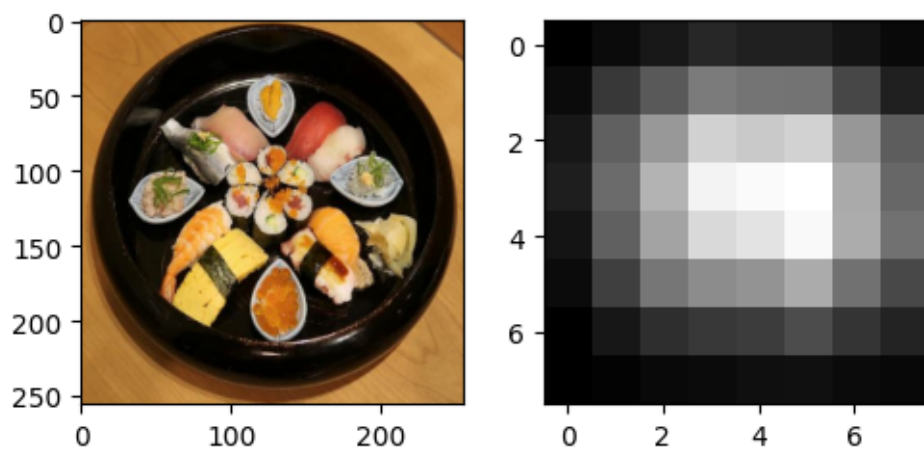
1/1 [=====] - 0s 23ms/step



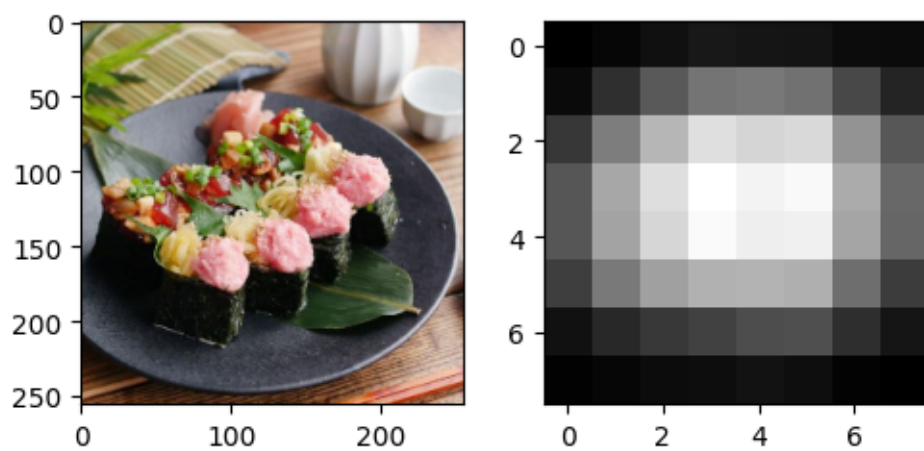
1/1 [=====] - 0s 24ms/step



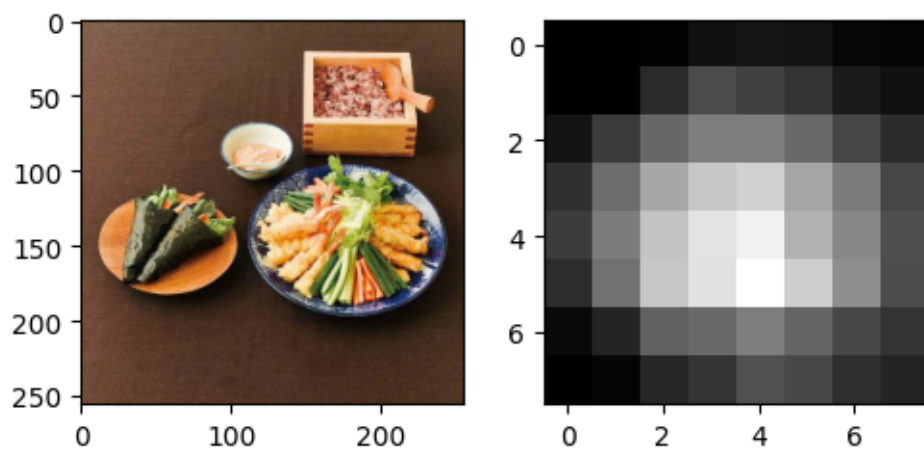
1/1 [=====] - 0s 25ms/step



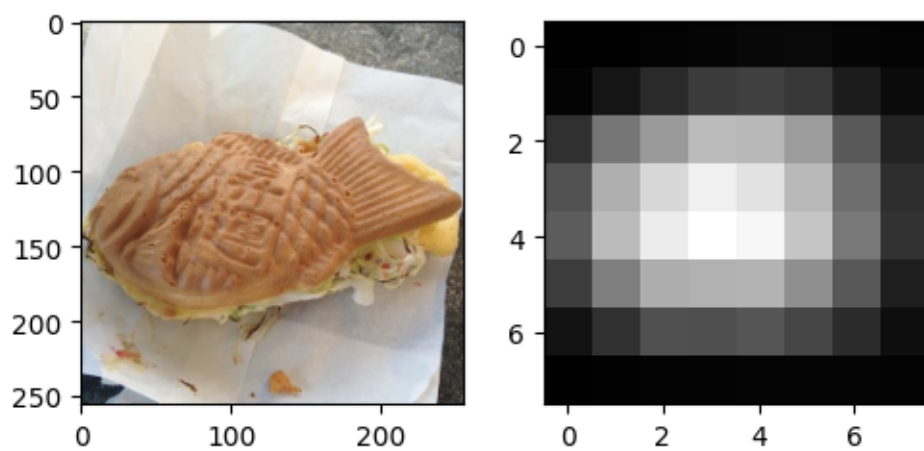
1/1 [=====] - 0s 23ms/step



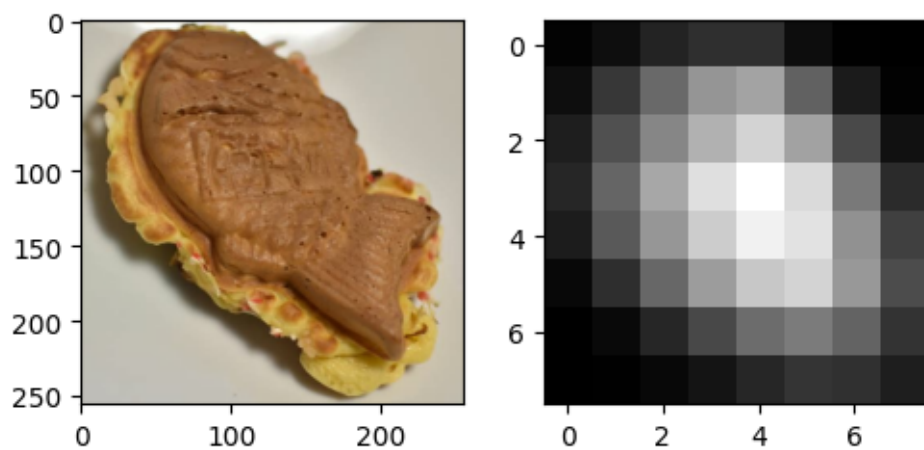
1/1 [=====] - 0s 24ms/step



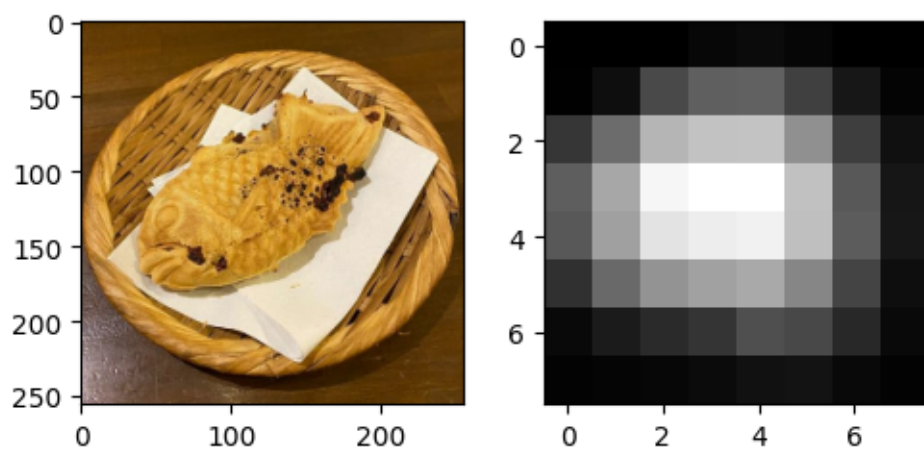
1/1 [=====] - 0s 29ms/step



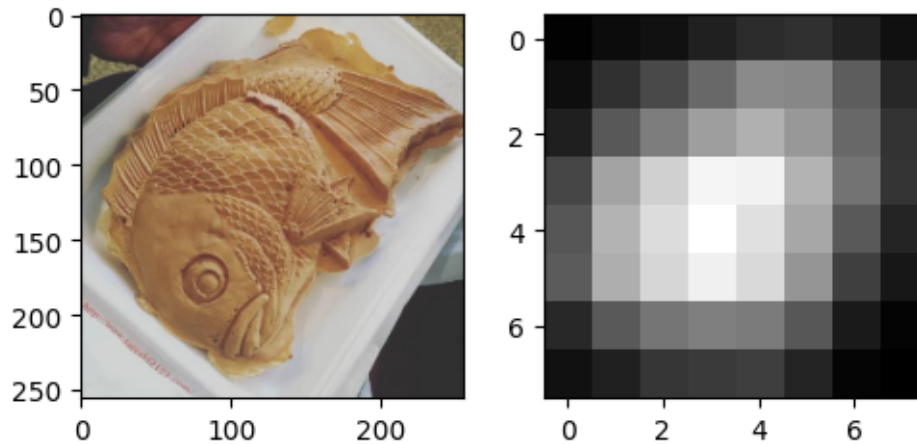
1/1 [=====] - 0s 22ms/step



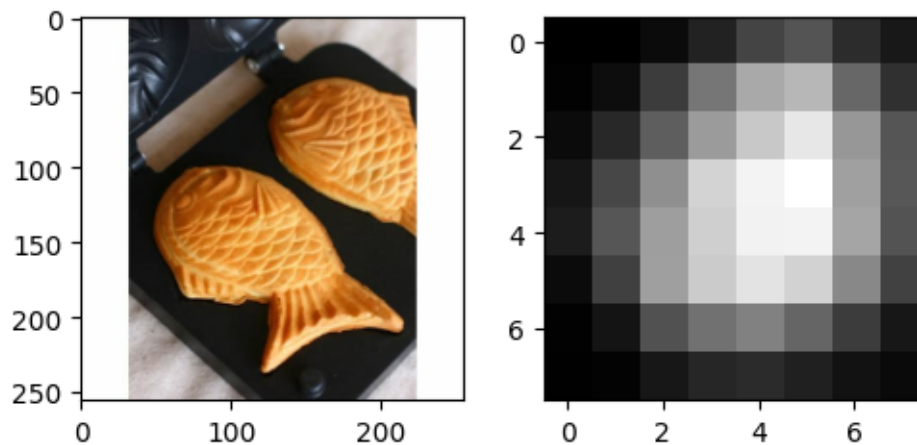
1/1 [=====] - 0s 22ms/step



1/1 [=====] - 0s 24ms/step



1/1 [=====] - 0s 23ms/step



Bardzo ciekawe wyniki.

Wygląda na to, że każda kategoria jest bardzo słusznie rozpoznawana, po rzeczywiście istotnych cechach (cechach jedzenia, a nie np. talerzy etc.).

Nawet podchwytliwe zdjęcie zawierające kompilację kilku zdjęć z sushi wypadło całkiem nieźle, choć zdecydowanie najgorzej ze wszystkich i jest to moim zdaniem zrozumiałe.

```
[ ]: wrongs = []

for images, labels in train_ds:
    bools = tf.math.argmax(labels,axis=1) != tf.math.argmax(model.
    ↪predict(images), axis=1)
    wrongs += list(zip(images[bools], labels[bools]))
```

```

for images, labels in test_ds:
    bools = tf.math.argmax(labels,axis=1) != tf.math.argmax(model.
↳predict(images), axis=1)
    wrongs += list(zip(images[bools], labels[bools]))

print(len(wrongs))

```

```

2/2 [=====] - 0s 325ms/step
2/2 [=====] - 0s 258ms/step
2/2 [=====] - 0s 260ms/step
2/2 [=====] - 0s 253ms/step
2/2 [=====] - 0s 254ms/step
2/2 [=====] - 0s 253ms/step
2/2 [=====] - 0s 253ms/step
2/2 [=====] - 0s 254ms/step
2/2 [=====] - 0s 255ms/step
2/2 [=====] - 0s 256ms/step
2/2 [=====] - 0s 261ms/step
2/2 [=====] - 0s 327ms/step
2/2 [=====] - 0s 261ms/step
2/2 [=====] - 0s 253ms/step
2/2 [=====] - 1s 881ms/step
3

```

Wydało mi się, że dataset shuffluje batche, dlatego niestety postaram się tę część zrobić bardziej interakcyjnie...

Na małym zbiorze danych nie powinno to być dużym problemem.

```

[ ]: def print_4(img, heat1, heat2, heat3):
    plt.figure(figsize=(15, 15))
    plt.subplot(5,5,1)
    plt.imshow(img)
    plt.subplot(5,5,2)
    plt.imshow(heat1, cmap='gray')
    plt.subplot(5,5,3)
    plt.imshow(heat2, cmap='gray')
    plt.subplot(5,5,4)
    plt.imshow(heat3, cmap='gray')
    plt.show()

[ ]: for image, label in wrongs:
    res = tmp_model.predict(tf.expand_dims(image, 0))[0]
    pred = model.predict(tf.expand_dims(image, 0))[0]
    print(f"Expected={class_names[tf.math.
↳argmax(label)]}\t\t\tPredicted={class_names[tf.math.
↳argmax(pred)]}\t\t\t{pred}")
    print_4(image,
        get_heat(res, model.layers[-1].get_weights()[0], 0),

```

```
get_heat(res, model.layers[-1].get_weights()[0], 1),
get_heat(res, model.layers[-1].get_weights()[0], 2))
```

1/1 [=====] - 0s 26ms/step

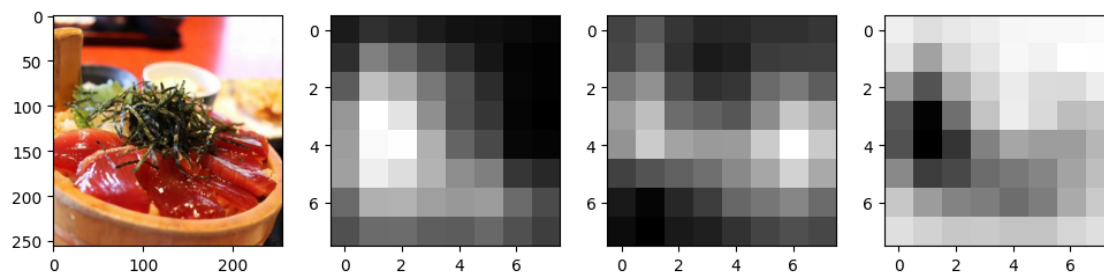
1/1 [=====] - 0s 25ms/step

Expected=sushi

Predicted=ramen

[0.6697369

0.29162917 0.03863393]



1/1 [=====] - 0s 23ms/step

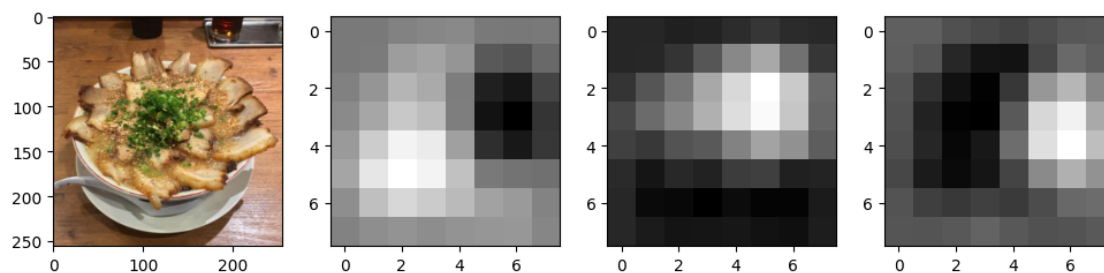
1/1 [=====] - 0s 25ms/step

Expected=ramen

Predicted=sushi

[0.3538958

0.45931664 0.18678764]



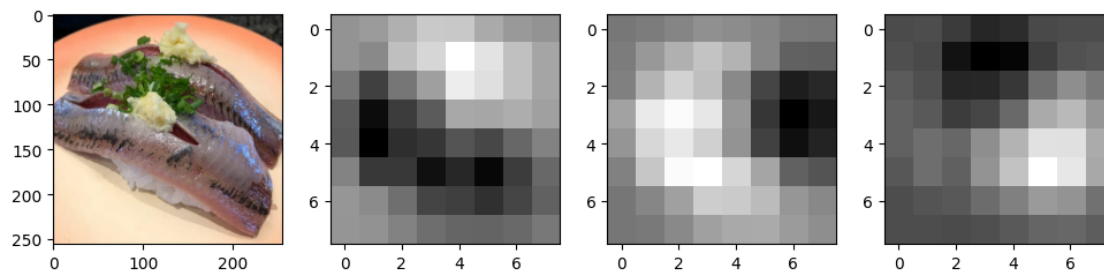
1/1 [=====] - 0s 23ms/step

1/1 [=====] - 0s 25ms/step

Expected=sushi

Predicted=taiyaki

[0.16951361 0.36088547 0.46960092]



Myślę, że pomijając drugi obrazek, przedstawiający ramen, w którym niewiele widać, pierwszy i trzeci człowiek by dobrze sklasyfikował.

Spośród 3 kategori, nawet z drugim obrazkiem człowiek by sobie poradził.

Z pewnością nie są to natomiast bardzo typowe elementy klasy (żaden z nich nie jest bardzo typowy, prawdopodobnie najbardziej typowy jest trzeci).

Jak widać, model nie ma dużej pewności co do żadnej klasy, aczkolwiek zazwyczaj ma widocznie klasę do której obrazek nie należy.

Po analizie heatmap, możemy stwierdzić, że model widzi różne cechy różnych klas i żadna nie wybija się znacząco ponad pozostałe klasy.

Cieężko mi coś więcej wywnioskować.