Feature Indexing for Perfume Bottles

Updates & Future Work

Chadi Barakat

JÜLICH Forschungszentrum | JÜLICH SUPERCOMPUTING CENTRE

UNIVERSITY OF ICELAND
FACULTY OF INDUSTRIAL ENGINEERING, MECHANICAL ENGINEERING AND COMPUTER SCIENCE

Initial Approach: Apply pre-trained Networks

- ResNet50 [1]

Apply ResNet50

In [2]:
```
model = ResNet50(input_shape=(224,224,3))
```

In [10]:
```
predictions = pd.DataFrame()
imgin = cv2.imread('products/0084.jpg')
imgin = cv2.cvtColor(imgin, cv2.COLOR_BGR2RGB)
image = np.reshape(imgin, newshape=(1,224,224,3))
preds = model.predict(image)
predictions = pd.DataFrame(decode_predictions(preds, top=10))
```

In [11]:
```
plt.imshow(imgin)
```

Out[11]: <matplotlib.image.AxesImage at 0x7fbcdb240990>

In [12]:
```
predictions
```

Out[12]:

	0	1	2	3	4	5	6	7	8	9
0	(n03916031, perfume, 0.9999505)	(n03690938, lotion, 1.5465565e-05)	(n04380533, table_lamp, 1.1884465e-05)	(n02948072, candle, 3.7579064e-06)	(n03314780, face_powder, 3.1802883e-06)	(n03666591, lighter, 1.3714757e-06)	(n03637318, lampshade, 1.0748321e-06)	(n03676483, lipstick, 9.390771e-07)	(n04357314, sunscreen, 9.062265e-07)	(n04286575, spotlight, 8.187738e-07)

In [50]:
```
predictions.index = data.index
```

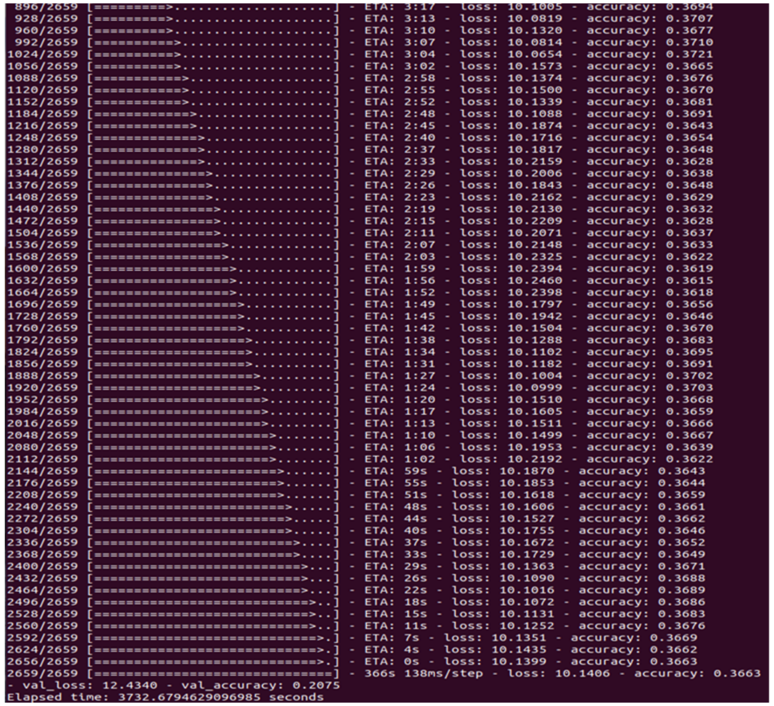
ResNet50 output

Second Approach: Retraining the Network

- Load ResNet50 without the output layer.

- Append a classification layer that is trained on selected outputs.

- Build the model



```
 896/2659 [=========>....................] - ETA: 3:17 - loss: 10.1005 - accuracy: 0.3694
 928/2659 [=========>....................] - ETA: 3:13 - loss: 10.0819 - accuracy: 0.3707
 960/2659 [=========>....................] - ETA: 3:10 - loss: 10.1320 - accuracy: 0.3677
 992/2659 [==========>...................] - ETA: 3:07 - loss: 10.0814 - accuracy: 0.3710
1024/2659 [==========>...................] - ETA: 3:04 - loss: 10.0654 - accuracy: 0.3721
1056/2659 [==========>...................] - ETA: 3:02 - loss: 10.1573 - accuracy: 0.3665
1088/2659 [===========>..................] - ETA: 2:58 - loss: 10.1374 - accuracy: 0.3676
1120/2659 [===========>..................] - ETA: 2:55 - loss: 10.1500 - accuracy: 0.3670
1152/2659 [===========>..................] - ETA: 2:52 - loss: 10.1339 - accuracy: 0.3681
1184/2659 [============>.................] - ETA: 2:48 - loss: 10.1088 - accuracy: 0.3691
1216/2659 [============>.................] - ETA: 2:45 - loss: 10.1874 - accuracy: 0.3643
1248/2659 [=============>................] - ETA: 2:40 - loss: 10.1716 - accuracy: 0.3654
1280/2659 [=============>................] - ETA: 2:37 - loss: 10.1817 - accuracy: 0.3648
1312/2659 [=============>................] - ETA: 2:33 - loss: 10.2159 - accuracy: 0.3628
1344/2659 [==============>...............] - ETA: 2:29 - loss: 10.2006 - accuracy: 0.3638
1376/2659 [==============>...............] - ETA: 2:26 - loss: 10.1843 - accuracy: 0.3648
1408/2659 [==============>...............] - ETA: 2:23 - loss: 10.2162 - accuracy: 0.3629
1440/2659 [===============>..............] - ETA: 2:19 - loss: 10.2130 - accuracy: 0.3632
1472/2659 [===============>..............] - ETA: 2:15 - loss: 10.2209 - accuracy: 0.3628
1504/2659 [===============>..............] - ETA: 2:11 - loss: 10.2071 - accuracy: 0.3637
1536/2659 [================>.............] - ETA: 2:07 - loss: 10.2148 - accuracy: 0.3633
1568/2659 [================>.............] - ETA: 2:03 - loss: 10.2325 - accuracy: 0.3622
1600/2659 [=================>............] - ETA: 1:59 - loss: 10.2394 - accuracy: 0.3619
1632/2659 [=================>............] - ETA: 1:56 - loss: 10.2460 - accuracy: 0.3615
1664/2659 [=================>............] - ETA: 1:52 - loss: 10.2398 - accuracy: 0.3618
1696/2659 [==================>...........] - ETA: 1:49 - loss: 10.1797 - accuracy: 0.3656
1728/2659 [==================>...........] - ETA: 1:45 - loss: 10.1942 - accuracy: 0.3646
1760/2659 [==================>...........] - ETA: 1:42 - loss: 10.1504 - accuracy: 0.3670
1792/2659 [===================>..........] - ETA: 1:38 - loss: 10.1288 - accuracy: 0.3683
1824/2659 [===================>..........] - ETA: 1:34 - loss: 10.1102 - accuracy: 0.3695
1856/2659 [===================>..........] - ETA: 1:31 - loss: 10.1182 - accuracy: 0.3691
1888/2659 [====================>.........] - ETA: 1:27 - loss: 10.1004 - accuracy: 0.3702
1920/2659 [====================>.........] - ETA: 1:24 - loss: 10.0999 - accuracy: 0.3703
1952/2659 [=====================>........] - ETA: 1:20 - loss: 10.1510 - accuracy: 0.3668
1984/2659 [=====================>........] - ETA: 1:17 - loss: 10.1605 - accuracy: 0.3659
2016/2659 [=====================>........] - ETA: 1:13 - loss: 10.1511 - accuracy: 0.3666
2048/2659 [======================>.......] - ETA: 1:10 - loss: 10.1499 - accuracy: 0.3667
2080/2659 [======================>.......] - ETA: 1:06 - loss: 10.1953 - accuracy: 0.3639
2112/2659 [======================>.......] - ETA: 1:02 - loss: 10.2192 - accuracy: 0.3622
2144/2659 [=======================>......] - ETA: 59s - loss: 10.1870 - accuracy: 0.3643
2176/2659 [=======================>......] - ETA: 55s - loss: 10.1853 - accuracy: 0.3644
2208/2659 [=======================>......] - ETA: 51s - loss: 10.1618 - accuracy: 0.3659
2240/2659 [========================>.....] - ETA: 48s - loss: 10.1606 - accuracy: 0.3661
2272/2659 [========================>.....] - ETA: 44s - loss: 10.1527 - accuracy: 0.3662
2304/2659 [========================>.....] - ETA: 40s - loss: 10.1755 - accuracy: 0.3646
2336/2659 [=========================>....] - ETA: 37s - loss: 10.1672 - accuracy: 0.3652
2368/2659 [=========================>....] - ETA: 33s - loss: 10.1729 - accuracy: 0.3649
2400/2659 [==========================>...] - ETA: 29s - loss: 10.1363 - accuracy: 0.3671
2432/2659 [==========================>...] - ETA: 26s - loss: 10.1090 - accuracy: 0.3688
2464/2659 [==========================>...] - ETA: 22s - loss: 10.1016 - accuracy: 0.3689
2496/2659 [===========================>..] - ETA: 18s - loss: 10.1072 - accuracy: 0.3686
2528/2659 [===========================>..] - ETA: 15s - loss: 10.1131 - accuracy: 0.3683
2560/2659 [===========================>..] - ETA: 11s - loss: 10.1252 - accuracy: 0.3676
2592/2659 [============================>.] - ETA: 7s - loss: 10.1351 - accuracy: 0.3669
2624/2659 [============================>.] - ETA: 4s - loss: 10.1435 - accuracy: 0.3662
2656/2659 [============================>.] - ETA: 0s - loss: 10.1399 - accuracy: 0.3663
2659/2659 [==============================] - 366s 138ms/step - loss: 10.1406 - accuracy: 0.3663
- val_loss: 12.4340 - val_accuracy: 0.2075
Elapsed time: 3732.6794629096985 seconds
```

Training on the Cluster

Improving the Results

- Provide better training labels.
- Split classification methods for improved detection:
  - Colour classification [Done]
  - Image adjustment [Current Work]
  - Shape identification [Current work]
    - Better training labels
    - Data augmentation

Improving the Results

- Provide better training labels.
- Split classification methods for improved detection:
  - Colour classification [Done]
  - Image adjustment [Current Work]
  - Shape identification [Current work]
    - Better training labels
    - Data augmentation

Improving the Results

- Provide better training labels.
- Split classification methods for improved detection:
    - Colour classification [Done]
    - Image adjustment [Current Work]
    - Shape identification [Current work]
        - Better training labels
        - Data augmentation

Improving the Results

- Provide better training labels.
- Split classification methods for improved detection:
  - Colour classification [Done]
  - Image adjustment [Current Work]
  - Shape identification [Current work]
    - Better training labels
    - Data augmentation

References

1. K. He, X. Zhang, S. Ren, J. Sun, "Deep Residual Learning for Image Recognition", 29th IEEE Conference on Computer Vision and Pattern Recognition, pp. 770-778, June 2016, doi:10.1109/CVPR.2016.90
2. P. Sermanet, D. Eigen, X. Zhang, M. Mathieu, R. Fergus, Y. Lecun, "OverFeat: Integrated Recognition, Localization and Detection using Convolutional Networks", International Conference on Learning Representations (ICLR), December 2013. [Online]. Available: http://arxiv.org/abs/1312.6229
3. https://www.python.org/
4. https://keras.io/
5. https://www.deep-projects.eu/