Project 4-1

Subject:

Attention Based CNN-ConvLSTM for Pedestrian Attribute Recognition

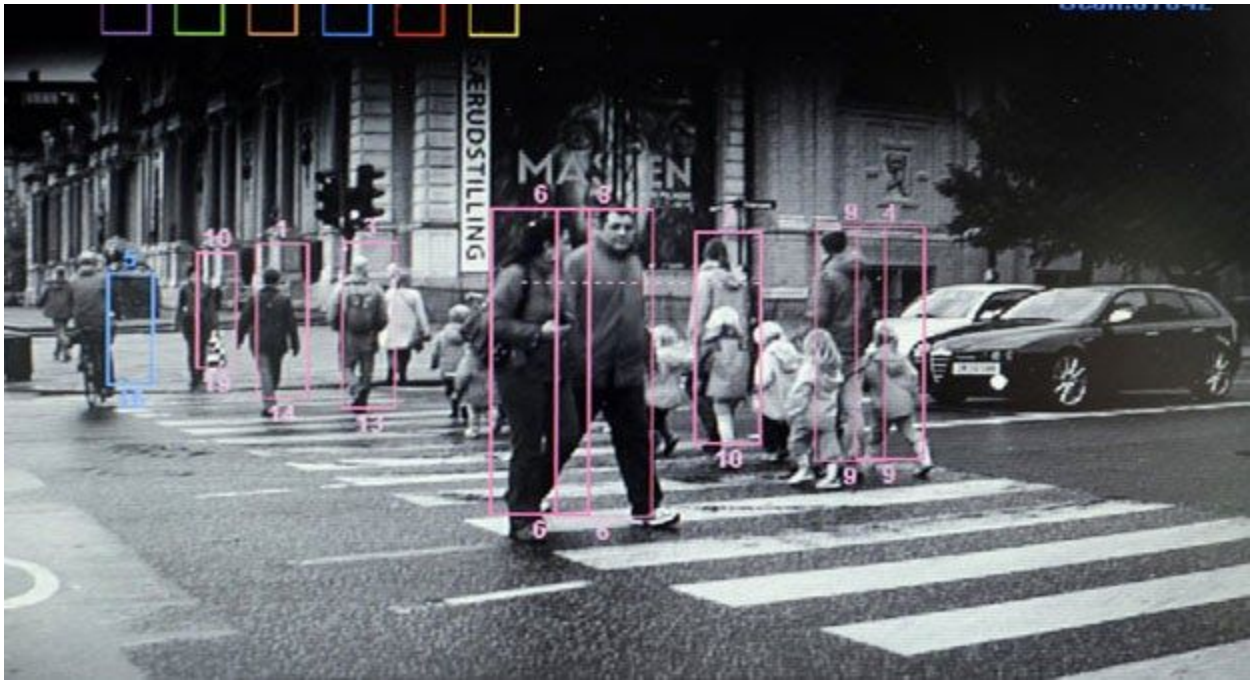
Authors:

Zahra MohammadBeigi

Student no. : 97222079

Ghazal Rafiei

Student no. : 97222044



Introduction

Recognizing pedestrian attributes is an important task in computer vision community due to it plays an important role in video surveillance.

Pedestrian attribution recognition is the task of recognising pedestrian features – such as whether they are female or male, whether they have a backpack, and so on.

Model:

Small filters did better than large ones ,although image features are big as how we see them.

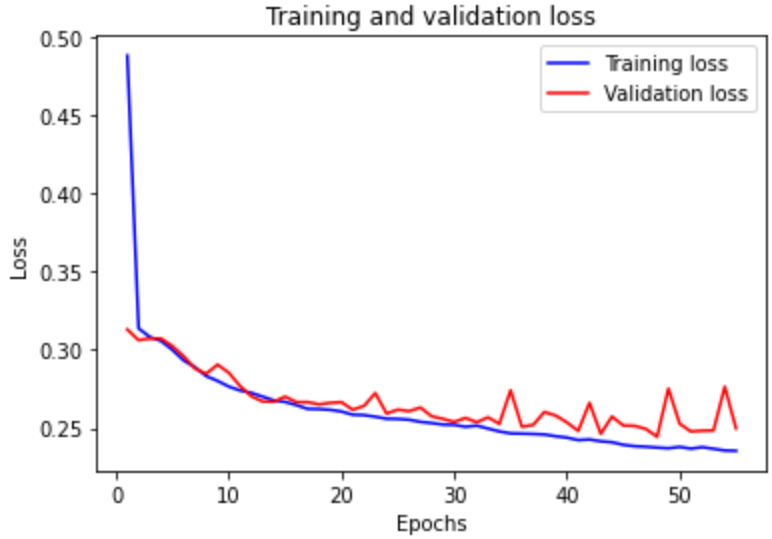
```
Model: "sequential"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
conv2d (Conv2D)              (None, 96, 96, 16)        1216
_________________________________________________________________
batch_normalization (BatchNo (None, 96, 96, 16)        64
_________________________________________________________________
max_pooling2d (MaxPooling2D) (None, 48, 48, 16)        0
_________________________________________________________________
dropout (Dropout)            (None, 48, 48, 16)        0
_________________________________________________________________
conv2d_1 (Conv2D)            (None, 44, 44, 32)        12832
_________________________________________________________________
max_pooling2d_1 (MaxPooling2 (None, 22, 22, 32)        0
_________________________________________________________________
batch_normalization_1 (Batch (None, 22, 22, 32)        128
_________________________________________________________________
dropout_1 (Dropout)          (None, 22, 22, 32)        0
_________________________________________________________________
conv2d_2 (Conv2D)            (None, 18, 18, 64)        51264
_________________________________________________________________
max_pooling2d_2 (MaxPooling2 (None, 9, 9, 64)          0
_________________________________________________________________
batch_normalization_2 (Batch (None, 9, 9, 64)          256
_________________________________________________________________
dropout_2 (Dropout)          (None, 9, 9, 64)          0
_________________________________________________________________
conv2d_3 (Conv2D)            (None, 5, 5, 64)          102464
_________________________________________________________________
max_pooling2d_3 (MaxPooling2 (None, 2, 2, 64)          0
_________________________________________________________________
batch_normalization_3 (Batch (None, 2, 2, 64)          256
_________________________________________________________________
dropout_3 (Dropout)          (None, 2, 2, 64)          0
_________________________________________________________________
```

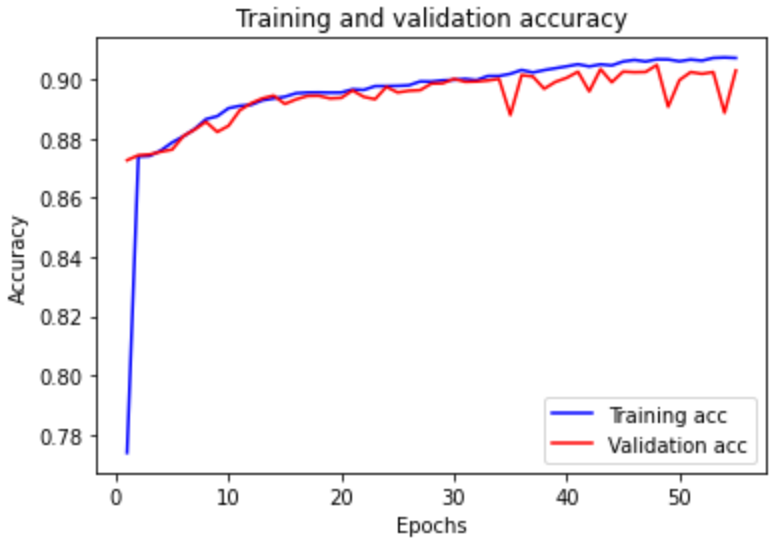
flatten (Flatten)	(None, 256)	0
dense (Dense)	(None, 128)	32896
dropout_4 (Dropout)	(None, 128)	0
dense_1 (Dense)	(None, 64)	8256
dropout_5 (Dropout)	(None, 64)	0
dense_2 (Dense)	(None, 92)	5980

Total params: 215,612
Trainable params: 215,260
Non-trainable params: 352

```
Epoch 1/100
33/33 [==============================] - 42s 1s/step - loss: 0.5935 - binary_accuracy: 0.6703 - val_loss: 0.3130 - val_binary_accuracy: 0.8728
Epoch 2/100
33/33 [==============================] - 40s 1s/step - loss: 0.3149 - binary_accuracy: 0.8744 - val_loss: 0.3062 - val_binary_accuracy: 0.8744
Epoch 3/100
33/33 [==============================] - 40s 1s/step - loss: 0.3073 - binary_accuracy: 0.8743 - val_loss: 0.3072 - val_binary_accuracy: 0.8747
Epoch 4/100
33/33 [==============================] - 40s 1s/step - loss: 0.3062 - binary_accuracy: 0.8753 - val_loss: 0.3072 - val_binary_accuracy: 0.8756
Epoch 5/100
33/33 [==============================] - 41s 1s/step - loss: 0.3014 - binary_accuracy: 0.8777 - val_loss: 0.3024 - val_binary_accuracy: 0.8764
Epoch 6/100
33/33 [==============================] - 40s 1s/step - loss: 0.2946 - binary_accuracy: 0.8805 - val_loss: 0.2959 - val_binary_accuracy: 0.8809
Epoch 7/100
33/33 [==============================] - 40s 1s/step - loss: 0.2882 - binary_accuracy: 0.8838 - val_loss: 0.2881 - val_binary_accuracy: 0.8831
Epoch 8/100
33/33 [==============================] - 40s 1s/step - loss: 0.2857 - binary_accuracy: 0.8853 - val_loss: 0.2847 - val_binary_accuracy: 0.8858
Epoch 9/100
33/33 [==============================] - 40s 1s/step - loss: 0.2797 - binary_accuracy: 0.8873 - val_loss: 0.2906 - val_binary_accuracy: 0.8823
Epoch 10/100
33/33 [==============================] - 40s 1s/step - loss: 0.2756 - binary_accuracy: 0.8905 - val_loss: 0.2854 - val_binary_accuracy: 0.8844
Epoch 11/100
33/33 [==============================] - 40s 1s/step - loss: 0.2765 - binary_accuracy: 0.8890 - val_loss: 0.2769 - val_binary_accuracy: 0.8897
Epoch 12/100
33/33 [==============================] - 40s 1s/step - loss: 0.2717 - binary_accuracy: 0.8917 - val_loss: 0.2704 - val_binary_accuracy: 0.8920
Epoch 13/100
33/33 [==============================] - 40s 1s/step - loss: 0.2671 - binary_accuracy: 0.8945 - val_loss: 0.2670 - val_binary_accuracy: 0.8937
Epoch 14/100
33/33 [==============================] - 40s 1s/step - loss: 0.2696 - binary_accuracy: 0.8927 - val_loss: 0.2669 - val_binary_accuracy: 0.8946
Epoch 15/100
33/33 [==============================] - 40s 1s/step - loss: 0.2663 - binary_accuracy: 0.8938 - val_loss: 0.2701 - val_binary_accuracy: 0.8918
Epoch 16/100
33/33 [==============================] - 40s 1s/step - loss: 0.2645 - binary_accuracy: 0.8956 - val_loss: 0.2666 - val_binary_accuracy: 0.8934
Epoch 17/100
33/33 [==============================] - 40s 1s/step - loss: 0.2618 - binary_accuracy: 0.8957 - val_loss: 0.2666 - val_binary_accuracy: 0.8945
Epoch 18/100
33/33 [==============================] - 40s 1s/step - loss: 0.2600 - binary_accuracy: 0.8961 - val_loss: 0.2652 - val_binary_accuracy: 0.8946

33/33 [==============================] - 40s 1s/step - loss: 0.2461 - binary_accuracy: 0.9028 - val_loss: 0.2603 - val_binary_accuracy: 0.8969
Epoch 39/100
33/33 [==============================] - 40s 1s/step - loss: 0.2437 - binary_accuracy: 0.9038 - val_loss: 0.2580 - val_binary_accuracy: 0.8993
Epoch 40/100
33/33 [==============================] - 40s 1s/step - loss: 0.2437 - binary_accuracy: 0.9045 - val_loss: 0.2537 - val_binary_accuracy: 0.9006
Epoch 41/100
33/33 [==============================] - 40s 1s/step - loss: 0.2403 - binary_accuracy: 0.9059 - val_loss: 0.2482 - val_binary_accuracy: 0.9027
Epoch 42/100
33/33 [==============================] - 40s 1s/step - loss: 0.2399 - binary_accuracy: 0.9049 - val_loss: 0.2661 - val_binary_accuracy: 0.8961
Epoch 43/100
33/33 [==============================] - 39s 1s/step - loss: 0.2429 - binary_accuracy: 0.9051 - val_loss: 0.2464 - val_binary_accuracy: 0.9036
Epoch 44/100
33/33 [==============================] - 40s 1s/step - loss: 0.2418 - binary_accuracy: 0.9040 - val_loss: 0.2574 - val_binary_accuracy: 0.8991
Epoch 45/100
33/33 [==============================] - 39s 1s/step - loss: 0.2363 - binary_accuracy: 0.9074 - val_loss: 0.2517 - val_binary_accuracy: 0.9027
Epoch 46/100
33/33 [==============================] - 39s 1s/step - loss: 0.2411 - binary_accuracy: 0.9060 - val_loss: 0.2514 - val_binary_accuracy: 0.9025
Epoch 47/100
33/33 [==============================] - 39s 1s/step - loss: 0.2368 - binary_accuracy: 0.9068 - val_loss: 0.2495 - val_binary_accuracy: 0.9026
Epoch 48/100
33/33 [==============================] - 40s 1s/step - loss: 0.2395 - binary_accuracy: 0.9060 - val_loss: 0.2445 - val_binary_accuracy: 0.9050
Epoch 49/100
33/33 [==============================] - 40s 1s/step - loss: 0.2370 - binary_accuracy: 0.9073 - val_loss: 0.2753 - val_binary_accuracy: 0.8909
Epoch 50/100
33/33 [==============================] - 40s 1s/step - loss: 0.2367 - binary_accuracy: 0.9069 - val_loss: 0.2528 - val_binary_accuracy: 0.8999
Epoch 51/100
33/33 [==============================] - 40s 1s/step - loss: 0.2380 - binary_accuracy: 0.9059 - val_loss: 0.2479 - val_binary_accuracy: 0.9026
Epoch 52/100
33/33 [==============================] - 39s 1s/step - loss: 0.2374 - binary_accuracy: 0.9062 - val_loss: 0.2483 - val_binary_accuracy: 0.9020
Epoch 53/100
33/33 [==============================] - 39s 1s/step - loss: 0.2393 - binary_accuracy: 0.9068 - val_loss: 0.2485 - val_binary_accuracy: 0.9026
Epoch 54/100
33/33 [==============================] - 39s 1s/step - loss: 0.2354 - binary_accuracy: 0.9078 - val_loss: 0.2764 - val_binary_accuracy: 0.8888
Epoch 55/100
33/33 [==============================] - 39s 1s/step - loss: 0.2344 - binary_accuracy: 0.9076 - val_loss: 0.2498 - val_binary_accuracy: 0.9031
Epoch 00055: early stopping
```





Does our network really perform well or it's just about sparsity of the labels matrix and the most elements are zero?

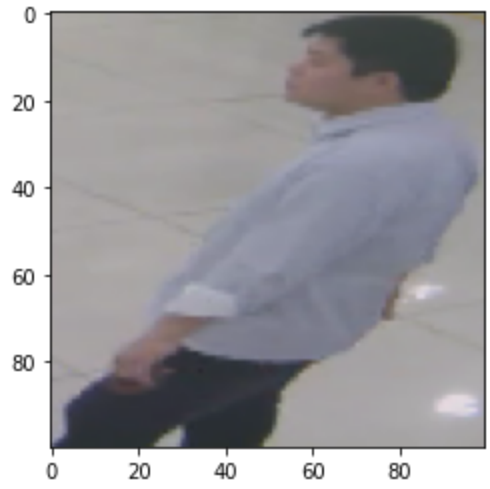
In order to check out, we set all y_train's elements to zero than then calculated the accuracy:

```
0.84684783
```

There is not much difference but we performed a little better! :))

Making a predict of an image in training set:

We chose the below picture randomly:



The first then predictions:

```
occlusion-Environment (0.993)
Clerk (0.927)
hs-Hat (0.917)
BodyThin (0.759)
up-Black (0.725)
BodyFat (0.579)
lb-Skirt (0.454)
low-White (0.451)
AgeLess16 (0.434)
up-White (0.38)
```

The model didn't do a good job on 'hs-Hat' and 'up-_Black' and 'up_white' But the other predictions sound good.