DIP-HW12

January 8, 2020

1 Digital Image Processing - HW12 - 98722278 - Mohammad Doosti Lakhani

In this notebook, I have solved the assignment's problems which are as follows:

1. Consider SSD300 and VGG16 models and answer following questions:
    1. What is the number of parameters in SSD300, number of multiplication and addition operations
    2. If we extract 2000 candidate regions from the input image and use VGG16 for classification, what is the number of Number of parameters in SSD300, number of multiplication and addition operations
    3. Compare results
2. `ground-truth.xlsx` and `detections.xlsx` files demonstrate ground truth and detected anchors respectively. Calculate `AP25` and `AP50`.
3. Train a model similar to SSD300 for object detection. *[THIS PART HAS BEEN IMPLEMENTED IN SEPARATE NOTEBOOK]*

1.1 1 Consider SSD300 and VGG16 models and answer following questions:

1. What is the number of parameters in SSD300, number of multiplication and addition operations
2. If we extract 2000 candidate regions from the input image and use VGG16 for classification, what is the number of Number of parameters in SSD300, number of multiplication and addition operations
3. Compare results

1.1.1 1.A Params and MAC in SSD300

Layer Name	# Params	MAC
conv1_1	1792	155520000
conv1_2	36928	3317760000
conv2_1	73856	1658880000
conv2_2	147584	3317760000
conv3_1	295168	1658880000
conv3_2	590080	3317760000

Layer Name	# Params	MAC
conv3_3	590080	3317760000
conv4_1	1180160	1703411712
conv4_2	2359808	3406823424
conv4_3	2359808	3406823424
conv4_3_norm	2	739328
conv4_3_norm_conf	387156	558931968
conv4_3_norm_loc	73744	106463232
conv5_1	2359808	851705856
conv5_2	2359808	851705856
conv5_3	2359808	851705856
fc6	4719616	1703411712
fc7	1049600	378535936
fc7_conf	1161342	419198976
fc7_loc	221208	79847424
conv6_1	262400	94633984
conv6_2	1180160	117964800
conv6_2_conf	580734	58060800
conv6_2_loc	110616	11059200
conv7_1	65664	6553600
conv7_2	295168	7372800
conv7_2_conf	290430	7257600
conv7_2_loc	55320	1382400
conv8_1	32896	819200
conv8_2	295168	2654208
conv8_2_conf	193620	1741824
conv8_2_loc	36880	331776
conv9_1	32896	294912
conv9_2	295168	294912
conv9_2_conf	193620	193536
conv9_2_loc	36880	36864

`Total Params = 26284974` `Total MACs = 31374277120`

Note: For naming convention, please see official SSD300 Caffe implementation or mine as separate file `DIP-HW12-Task3`.

1.1.2 1.B Params and MAC in VGG16 for 2000 Candidates

Layer Name	# Params	MAC
conv1_1	1792	86704128
conv1_2	36928	1849688064
conv2_1	73856	924844032
conv2_2	147584	1849688064
conv3_1	295168	924844032
conv3_2	590080	1849688064
conv3_3	590080	1849688064

Layer Name	# Params	MAC
conv4_1	1180160	924844032
conv4_2	2359808	1849688064
conv4_3	2359808	1849688064
conv5_1	2359808	462422016
conv5_2	2359808	462422016
conv5_3	2359808	462422016
fc6	102764544	102760448
fc7	16781312	16777216
fc8	4097000	4096000

```
Total Params for SINGLE VGG16 RUN = 138357544    Total MACs for SINGLE VGG16 RUN =
15470264320
   Total Params for 2000 VGG16 RUN = 138357544     Total MACs for 2000 VGG16 RUN =
30940528640000
```

1.1.3 1.C Compare Results

Model Name	# Params	MAC
VGG16-2000	138357544	30940528640000
VGG16	138357544	15470264320
SSD300	26284974	31374277120

As we can see, even though SSD has VGG in itself plus much more conv layers across the network, it has much less parameters because of omitting fully connected layers and replacing them with conv layers where the ratio is about `5.26`, so SSD300 is fifth of VGG16.

About number of multiplications and additions for SINGLE run of VGG16, we need about `0.493` ratio of this amount in SSD300 so still even SSD300 is much bigger in term of number of layers, it reasonable number of m-a operations but this time number of operations cannot be reduced by using conv layers instead of fc layers as we can see the number of operations layer by layer in above sections.

But as we want to run VGG16 for 2000 candidate regions, we need about `9.861` times more m-a operations in VGG16 w.r.t. SSD300 which has immeensly increased and this is the main reason that networks like RCNN are very slow and fail to operate in real time tasks.

1.2 2. `AP25` and `AP50` for `ground-truth.xlsx` and `detections.xlsx`

```
In [1]: import pandas as pd
        import numpy as np
```

```
In [86]: pred = pd.read_csv('pred.txt', sep='\t').sort_values(by='score', ascending=False)
         print(pred)
         pred = pred.to_numpy()
```

```
     x    y   w   h  score
3  114   31  14  21   0.96
```

```
2   55   72  34  36   0.89
0   11    5  19  26   0.84
1   18   39  31  23   0.79
6  124  136  29  35   0.74
4   24   98  21  34   0.47
5   36  150  41  26   0.39
7   92  153  27  47   0.29
```

In [87]:
```
gt = pd.read_csv('gt.txt', sep='\t')
print(gt)
gt = gt.to_numpy()
```

```
     x    y   w   h
0  153   21  20  25
1  116   30  13  23
2  125  135  30  35
3   30  160  30  20
4   10    5  20  25
```

In [94]:
```
threshold = 0.25
ious = []
for i in range(len(pred)):
    for j in range(len(gt)):
        xmin_max = max(gt[j, 0], pred[i, 0])
        ymin_max = max(gt[j, 1], pred[i, 1])
        xmax_min = min((gt[j, 0] + gt[j, 2]), (pred[i, 0] + pred[i, 2]))
        ymax_min = min((gt[j, 1] + gt[j, 3]), (pred[i, 1] + pred[i, 3]))
        intersection_area = max(0, xmax_min - xmin_max + 1) * max(0, ymax_min - ymin_
        gt_area = ((gt[j, 0] + gt[j, 2]) -gt[j, 0] + 1) * ((gt[j, 1] + gt[j, 3]) - gt
        pred_area = ((pred[i, 0] + pred[i, 2]) - pred[i, 0] + 1) * ((pred[i, 1] + pred
        union_area = gt_area + pred_area - intersection_area
        iou = intersection_area / union_area
        ious.append(iou)
ious = np.array(ious)
tp = len((ious > threshold).nonzero()[0])
fn = len(gt) - tp
fp = len(pred) - tp
precision = tp/(tp+fp)
recall = tp/(tp+fn)

pr = []

for recall_level in np.linspace(0.0, 1.0, 11):
    if recall >= recall_level:
        pr.append(precision)
    else:
```

```
                pr.append(0)
            avg_pr = np.mean(pr)
            print('AP is ', avg_pr)
            print('mAP is ', avg_pr)

            # please read blow section
```

```
AP is  0.409090909091
mAP is  0.409090909091
```

1. mAP50:
    1. AP is 0.204545454545
    2. mAP is 0.204545454545
2. mAP25:
    1. AP is 0.409090909091
    2. mAP is 0.409090909091

In the given assignment for task 2: 1. it has been said that these prediction and ground truth coordinates are for a SINGLE image. 2. the type of objects in this single image has not been declared so I considered it SINGLE TYPE OBJECT multiple instance mode.

Based on the reason 1, `mAP = mP` as there is only one image so average over all precision and recalls are the precision and recall of the single image.

Based on the reason 2, `mAP = AP` as there is only one object so `m` which stands for multi class objects for averaging is still one single number as there is only 1 object but multiple instance of it.