Calculation of mAP

A. mAP_processing.py

1. 介紹

mAP_processing.py 是一支對利用機器和不同model 所產生的ssd_log 或yolo_log 進行格式轉換的程式,目的是將其內容轉換為pascalvoc.py 所需的輸入項,將會存放於路徑下生成之GTLOG和AlLOG資料來中。

若為Auto模式則會自動執行pascalvoc.py,不需修改到pascalvoc.py之參數。

2. 操作說明

步驟1. 輸入所需之input

mAP_processing.py 進行格式轉換的輸入項包含:log_path_GT, log_path_Al 和pic_path 三者,執行的指令如下所示。

若希望轉檔後,直接執行pascalvoc.py ,則可以執行auto 模式

```
python3 mAP_processing.py \
--gt_path {GT xml folder} --ai_path {ssd_log or yolo_log .txt} --pic_path {image folder} \
--auto
```

步驟2. 取得轉檔後的檔案

mAP_processing.py 轉檔完成後,會將相同數量跟檔名的檔案分別以txt 的格式存放於GTLOG 和AILOG 兩個資料夾當中。

3. Example:

狀況1:以一般模式完成對GT_log 和AI_log 的轉檔工作

假設執行mAP_processing.py 所需的檔案路徑如下所示,並且僅先取得轉檔後的檔案,沒有要同步執行 pascalvoc.py 。

```
gt_path: ../example/GT/20201104-part2
ai_path: ../example/logs/20201104-part2_ssd_v5.1.1_640x360_Data20210420-retrain5b-
160k.txt
Pic path: ../example/pics/20201104-part2
```

步驟1. 輸入所需之input

操作指令如下所示。

```
python3 mAP_processing.py \
--gt_path ../example/GT/20201104-part2 \
--ai_path ../example/logs/20201104-part2_ssd_v5.1.1_640x360_Data20210420-
retrain5b-160k.txt \
--pic_path ../example/pics/20201104-part2
```

```
tools/AI_Evaluation$ python3 mAP_processing.py \
   --gt_path ../example/GT/20201104-part2 \
--ai_path ../example/logs/20201104-part2_ssd_v5.1.1_640x360_Data20210420-retrain5b-160k.txt \
--pic_path ../example/pics/20201104-part2
Load log from ../example/logs/20201104-part2
Load log from ../example/logs/20201104-part2_ssd_v5.1.1_640x360_Data20210420-retrain5b-160k.txt.
Load pics from ../example/pics/20201104-part2.
       === loadAItxt from ssd ======
100%|
                                                                                                                                | 251/251 [00:00<00:00, 15090.24it/s]
   ===== transformDict ======
   ===== rewrite AILOG ======
100%|
                                                                                                                                 | 251/251 [00:00<00:00, 1149.86it/s]
          Read xml files from ../example/GT/20201104-part2 ====
                                                                                                                                 | 251/251 [00:00<00:00, 5329.54it/s]
100%1
       === transformDict ======
        == rewrite GTLOG ======
100%|
                                                                                                                                 | 251/251 [00:00<00:00, 2457.39it/s]
        == Comparison AILOG File With GTLOG File ======
                                                                                                                                   | 251/251 [00:00<00:00, 356.57it/s]
```

步驟2. 取得轉檔後的GT_log 和AI_log 檔案

轉檔完成之檔案存放於GTLOG 和AILOG 兩個資料夾,可以看到兩個資料夾當中的檔案數量和檔名是相同的。

```
kaijiedeMacBook-Pro-2 pmx_tools/AI_Evaluation <master</pre>
                                                                                  *> » ls
 AILOG
GTLOG
                                                                                                           outPut
                                                                                     MBATool2.2.1.py pascalvoc.py
                                                      convertSSDLogfromRaw_MergeVersion.py
 README md
                                                     convertsables
countruntime.py
i=PrecisionRecall.py
 autoCreateJPG.py
                                                                                                         vertex
                                                                                                          writeXML.py
 convertIMG2BIN_MergeVersion.py
                                                     mAP_processing.py
20201104-part2_00037.txt 20201104-part2_00107.txt 20201104-part2_00177.txt 20201104-part2_00240.txt 20201104-part2_00047.txt 20201104-part2_00117.txt 20201104-part2_00180.txt 20201104-part2_00247.txt 20201104-part2_00157.txt 20201104-part2_00157.txt 20201104-part2_00157.txt 20201104-part2_00157.txt 20201104-part2_00157.txt 20201104-part2_00157.txt 20201104-part2_00157.txt
20201104-part2_00060.txt 20201104-part2_00127.txt 20201104-part2_00197.txt
```

狀況2:以auto 模式於完成轉檔的同時執行pascalvoc.py

步驟1. 輸入所需之input

操作指令如下所示。

```
python3 mAP_processing.py \
    --gt_path ../example/GT/20201104-part2 \
    --ai_path ../example/logs/20201104-part2_ssd_v5.1.1_640x360_Data20210420-
retrain5b-160k.txt \
    --pic_path ../example/pics/20201104-part2 \
    --auto
```

步驟2. 取得轉檔成功的檔案和mAP 的計算結果

可看見在auto 模式下,mAP_processing.py 會直接執行計算mAP 的pascalvoc.py ,並取得紅框中的計算結果。關於pascalvoc.py 的用途與操作方式請參考下段的介紹。

B. pascalvoc.py

1. 介紹

pascalvoc.py 是一支用於計算mAP之程式,取用自<u>rafaelpadilla 的github</u>。若要了解mAP 的基礎概念,可點選此處。

2. 操作說明

pascalvoc.py 的主要輸入項包含轉檔完成後的GT_log 和AI_log ,以及對於計算mAP 有影響的 Threshold 值,對於pascalvoc.py 運行的相關參數可以參考下圖。

```
This project applies the most popular metrics used to evaluate object detection algorithms. The current implemention runs the Pascal VOC metrics. For further references, please check: https://github.com/rafaelpadilla/Object-Detection-
Metrics
optional arguments:
                                           show this help message and exit
show program's version number and exit
folder containing your ground truth bounding boxes
folder containing your detected bounding boxes
IOU threshold. Default 0.5
    -h, --help
-v, --version
    -gt , --gt_path
-ai , --ai_path
-t , --threshold
                                            IOU threshold. Default 0.5

format of the coordinates of the ground truth bounding boxes: ('xywh': <left> <top> <width> <height>) or ('xyrb': <left> <top> <right> <bottom>)

format of the coordinates of the detected bounding boxes ('xywh': <left> <top> <width> <height>) or ('xyrb': <left> <top> <right> <bottom>)

reference of the ground truth bounding box coordinates: absolute values ('abs') or relative to its image size ('rel')

reference of the ground truth bounding box coordinates: absolute values ('abs') or relative to its image size ('rel')

image size (Required if -gtcoords or -detcoords are 'rel'

folder where the plots are saved
     -gtformat
    -detformat
    -gtcoords
    -detcoords
     -imgsize
             . --savepath
                                             folder where the plots are saved
    -np, --noplot
                                            no plot is shown during execution
Developed by: Rafael Padilla (rafael.padilla@smt.ufrj.br)
```

步驟1. 輸入相關參數

我們計算自機器所取得成果的mAP 時,Threshold 值固定為0.2 ,並採用不顯示圖片的方式,操作指令如下。

```
python pascalvoc.py -gt {轉檔完成之GT_log path} -ai {轉檔完成之AI_log path} -t 0.2 -np
```

步驟2. 取得mAP 的計算結果

計算mAP 所取得的各類別折線圖,會存放於results 資料夾當中,並依類別單張存取。

3. Example: 以pascalvoc.py 計算mAP

假設執行pascalvoc.py 所需的檔案路徑如下所示,且Threshold 值同樣為0.2 ,並同樣採用不顯示圖片的方式。

轉檔完成之GT_log path: ./GTLOG/ 轉檔完成之Al log path: ./AlLOG/

步驟1. 輸入相關參數

操作指令如下。

python pascalvoc.py -gt ./GTLOG/ -ai ./AILOG/ -t 0.2 -np

步驟2. 取得mAP 的計算結果

```
huangkaijiedeMacBook-Pro-2 pmx_tools/AI_Evaluation master*> > python3 pascalvoc.py -gt ./GTLOG/ -ai ./AILOG/ -t 0.2 -np
Folder /Users/chester.huang/Desktop/Git/pmx_tools/AI_Evaluation/results already exists and may contain important results.

Enter 'Y' to continue. WARNING: THIS WILL REMOVE ALL THE CONTENTS OF THE FOLDER!
Or enter 'N' to abort and choose another folder to save the results.

Y
AP: 85.10% (Bike)
AP: 60.89% (Person)
AP: 82.53% (Vehicle)
mAP: 76.17%
huangkaijiedeMacBook-Pro-2 pmx_tools/AI_Evaluation master*> > ls results
Bike.png Person.png Vehicle.png results.txt
```