OCR Project

By Zhuzhen Li

Introduction:

The aim of the project is to develop an OCR software application which can be used to correctly recognize the contents on a seven segment display screen in real time. Specifically, the users supposes to use this application to generate dataset, train the deep learning network, detect and recognize the contents, with achieving maximum accuracy and speed at the same time.

Object detection is one of the most core and challenging problem in the field of computer vision. The differences in shape, size, posture of each object, plus the interference of illumination and occlusion, this object detection task is to find all seven segment display numbers, locate and classify them correctly in real time video.

Background:

Problems to solve:

- Objects will display in any position
- Objects will be in different sizes
- The real-time detection process requires at least 10fps
- There will be multi-row of objects to detect
- There might be other characters other than 0-9 digits, such as: . - :

2 Available Approaches:

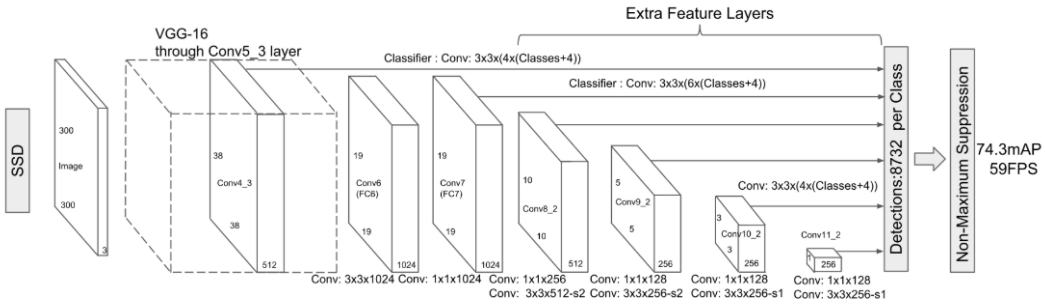
- **Two stages:**
  - Feature extraction ----> region proposal -----> classification
  - Common algorithm : R-CNN、SPP-Net、Fast R-CNN、Faster R-CNN, R-FCN

- **One stage**
  - Feature extraction ----> classification
  - Common algorithm: OverFeat、YOLOv1、YOLOv2、YOLOv3、SSD, RetinaNet

Implementation:

SSD:

- Based on the Anchor in Faster R-CNN, SSD brings similar Prior box.
- Predict with different scale feature maps, so that objects with different sizes can be detected
- VGG16 Network is used as feature extraction



Project implementation:

STEP1: Dataset preparation:

- Generate customized dataset

```
cd ocr-ledsegment-bitmap/text_gen/TextRecognitionDataGenerator
    python run.py -f 64 -l num -c 8000 --length 8 -k 5 -wd -5 -tc '#000000','#999999' -m
  20,20,20,20 -bl 0
    python gen_voc_ann.py
```

- Make sure here is no VOC2019 dataset already exist

```
cd ocr-ledsegment-bitmap/
rm -r VOC2019
```

- Generate VOC dataset

```
    python make_voc.py
    python make_train_val_test.py
```

STEP2: Pass dataset to Network:

- Make sure here is no VOC2019 dataset already exist

```
cd gluon-cv/docs/tutorials/detection/
    rm -r VOC2019
cp -r ~/ocr-ledsegment-bitmaps/VOC2019/ ./
```

STEP3: Training (on gpu):

```
source /opt/intel/compilers_and_libraries/linux/bin/compilervars.sh intel64
export PYTHONPATH=~/MXNet-MKL-DNN/incubator-mxnet/python/

rm -r number_train_result/
python train_ssd.py --save-prefix number_train_result2/ --epochs 70 --batch-size 32 --lr
0.001 --lr-decay 0.1 --lr-decay-epoch 40,50,60
```

Please note that the gluoncv-copy directory pulled from this repo is supposed be parallel with this ocr project directory.

Evaluation:

- Successfully generate the dataset in the appropriate format for training the deep learning network.
- Fine Tune, train and Inference the SSD-VGG network with result of the training mAP 95%+.
- Make the flow of generating dataset, training the deep learning network, detecting and recognizing contents on screen to be automatically and user friendly. In addition, the users can make their own desired dataset for similar training purpose with different script arguments.

Future improvement:

- Characters such as . : - could be added into dataset
- Current speed of inference is 1 s / image, which supposes to have a significant improvement
- SSD-VGG16 is currently used in this project, other network could be further tried and compared with each other on both accuracy and speed