

main

December 9, 2018

1 CM3: Convolution with Max Operator and Mathematical Morphology

This code is public available in: <https://github.com/falreis/segmentation-eval>

1.1 Activate Conda Environment

To create the development environment, create and start the Anaconda environment, that contains all software needed to run this notebook properly.

Below is the terminal scripts to create and run the Anaconda environment.

```
conda env create -f ../i2dl.yml
source activate i2dl
```

1.2 Download KITTI Road Dataset

Download must be done using browser, because it's necessary to inform your current mail to download the file.

Link: http://www.cvlibs.net/download.php?file=data_road.zip

1.2.1 Unzip the files

Put the *data_road* zip file inside the folder *datasets/Kitti/* and unzip the file using the command below.

```
unzip data_road.zip
```

1.2.2 Datasets Folder

The dataset folder should have a structure like this: - datasets/ - Kitti/ - data_road/ - testing/ - training/

1.3 Speed up training

To provide more speed in the training and testing operations, we put all images in Numpy array and read the images once. This operation avoid multiple disk/ssd readings.

To create it it's possible to use the command shell in AMRP folder or to use the notebook. Below the instructions to run with the command shell.

```
python kitti.py --func=npy --set=train --augm=true
```

1.3.1 Params

- set = {train | | test}
 - Parâmetro para conjunto de treino ou teste a ser utilizado.
- augm? = {false | | true}
 - Parâmetro para uso de dataset com data augmentation.

```
In [1]: import npy_kitti as nk
        nk.npy(set_name='train', augm=False)

../datasets/Kitti/data_road/training/image_2/
../datasets/Kitti/data_road/training/gt_image_2/
Done/ 289
```

1.4 Train the Network

To run the neural network, use the command shell with the scripts below.

```
python kitti.py --func=train --net=full --check=true --augm=true --load=true --epochs=100 --ba
```

```
python kitti.py --func=train --net=hed --merge=add --check=true --augm=true --load=true --epoch
```

```
python kitti.py --func=train --net=hed --merge=avg --check=true --augm=true --load=true --epoch
```

```
python kitti.py --func=train --net=hed --merge=max --check=true --augm=true --load=true --epoch
```

```
python kitti.py --func=train --net=hed --merge=maj --check=true --augm=true --load=true --epoch
```

1.4.1 Params

- func = { run }
 - Opção para execução da rede neural.
- net = { hed | | full }
 - Parâmetro correspondente ao nome da rede neural a ser utilizada (hed ou full).

- `merge = { avg || max || add || maj }`
 - Parâmetro correspondente ao nome da operação a ser executada.
- `check = {false || true}`
 - Parâmetro para armazenamento de pontos de controle.
- `vote? = {1-5} (OPCIONAL: somente para merge==maj)`
 - Parâmetro correspondente ao número de votos necessários para que sejam consideradas as maiorias.
- `augm? = {false || true} (OPCIONAL)`
 - Parâmetro para uso de dataset com data augmentation.
- `load? = {false || true} (OPCIONAL)`
 - Parâmetro para uso de pesos da rede vggnet16.
- `epochs? = {1-INF} (OPCIONAL)`
 - Parâmetro para definição do número de épocas para treinamento
- `balanced? = {false || true} (OPCIONAL)`
 - Parâmetro para definição de utilização de função *categorical_crossentropy* ou *balanced_categorical_crossentropy* (própria)
- `morf? = {false || true} (UNAVAILABLE)`
 - Parâmetro para aplicação de morfologia matemática a predição da rede neural.

```
In [ ]: import model_kitti as mk
import train_kitti as trk

model = mk.model_hed(merge_name='max', vote_value=None, morf=False)
trk.train(model=model, net='hed' merge='max', check=True, augm=False, load=True, nb_ep

In [ ]: import model_kitti as mk
import train_kitti as trk

model = mk.model_full()
trk.train(model=model, net='full' merge=None, check=True, augm=False, load=True, nb_ep
```

1.5 Test the Results

To test the networks, marking the road in the images or produce the ground-truths to submit into Kitti dataset, use the scrips below.

```
python kitti.py --func=test --net=hed --merge=max --set=train --mark=false --morf=true
```

```
python kitti.py --func=test --net=full --set=train --mark=false --morf=true
```

1.5.1 Params

- `net = { hed || full }`
 - Parâmetro correspondente ao nome da rede neural a ser utilizada.
- `merge = { avg || max || add || maj }`
 - Parâmetro correspondente ao nome da operação a ser executada.
- `set = {train || test}`
 - Parâmetro para conjunto de treino ou teste a ser utilizado.
- `mark = {false || true}`
 - Parâmetro para marcar rodovia sobre a imagem original ou gerar grond-truth.
- `morf? = {false || true}` (*OPCIONAL*)
 - Parâmetro para aplicação de morfologia matemática a predição da rede neural.

```
In [3]: #marcar rodovia sobre a imagem original
import model_kitti as mk
import test_kitti as tek

model = mk.model_hed(merge_name='max', vote_value=None, morf=False)
tek.test(model=model, net='hed', merge_name='max', set_name='test', mark=True, morf=False)
```

Using TensorFlow backend.

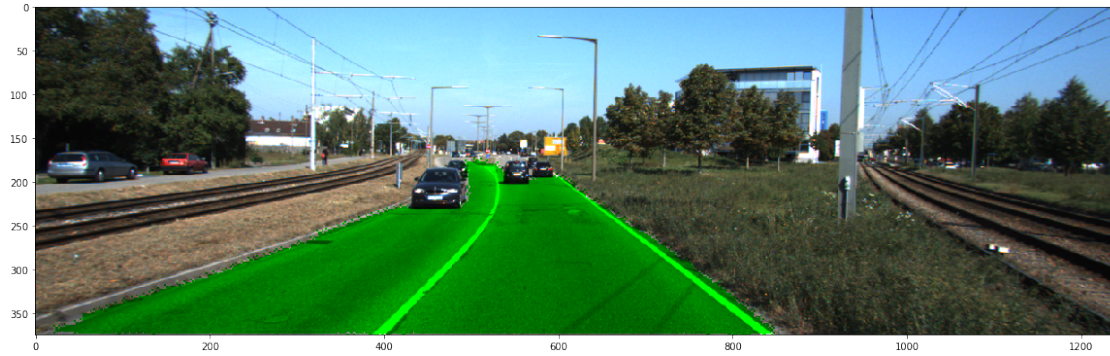
```
max
../weights/5k/hed_kitti_weight_max.best.hdf5
../export/Kitti/hed/test/
../datasets/Kitti/data_road/testing/image_2/*.
Done/ 290
```

```
In [15]: import matplotlib.pyplot as plt
import matplotlib.image as mpimg

img_name = '../export/Kitti/hed/test_mark/um_000005.png'

img = mpimg.imread(img_name)
plt.figure(figsize=(20,20))
plt.imshow(img)
```

```
Out[15]: <matplotlib.image.AxesImage at 0x7f77d370a2b0>
```



In [6]: *#mostrar a imagem resultante ("ground-truth" gerado)*

```
import warnings
```

```
warnings.filterwarnings("ignore")
```

```
import model_kitti as mk
```

```
import test_kitti as tek
```

```
model = mk.model_hed(merge_name='max', vote_value=None, morf=False)
```

```
tek.test(model=model, net='hed', merge_name='max', set_name='test', mark=False, morf=True)
```

max

```
../weights/5k/hed_kitti_weight_max.best.hdf5
```

```
../export/Kitti/hed/test/
```

```
../datasets/Kitti/data_road/testing/image_2/*.*
```

Done/ 290

In [16]: `import matplotlib.pyplot as plt`

```
import matplotlib.image as mpimg
```

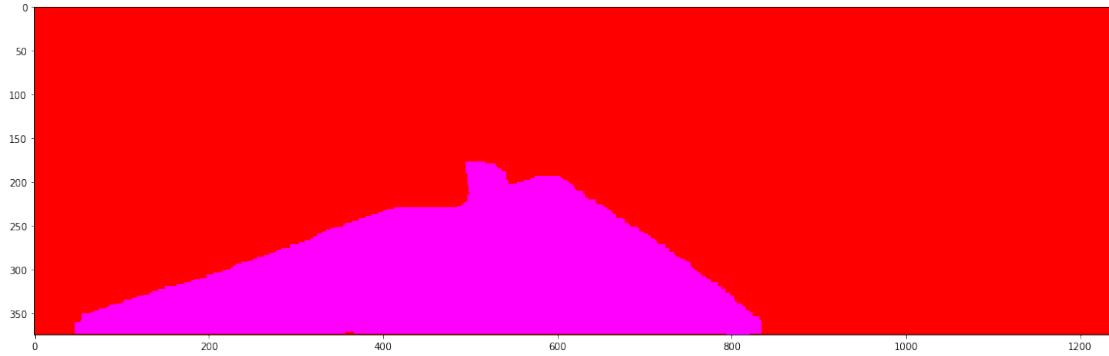
```
img_name = '../export/Kitti/hed/test/um_000005.png'
```

```
img = mpimg.imread(img_name)
```

```
plt.figure(figsize=(20,20))
```

```
plt.imshow(img)
```

Out[16]: `<matplotlib.image.AxesImage at 0x7f77d36e69e8>`



1.6 Evaluate Results

To eval the results, use the Kitti benchmark toolkit. The repository contains an version of Kitti benchmark toolkit with all pre-requisites into an Anaconda environment.

1.6.1 KITTI Eval Environment

To create the eval environment, run the commands below in the terminal.

```
conda env create -f ../../eval/kitti.yml
source activate kitti
```

1.6.2 Change File Names

To eval the results, change the name of the files create by the test procedure. Following this tutorial, it is necessary to install *rename* program, as the code below.

```
sudo apt install rename
cd ../export/Kitti/hed/train
rename 's/umm_/umm_road/' *
rename 's/um_/um_lane/' *
rename 's/uu_/uu_road/' *
```

1.6.3 Eval Procedure

The commands should have the same structure as the code below.

```
cd ../../eval/Kitti/
python evaluateRoad.py ../../code/export/Kitti/hed/train ../../code/datasets/Kitti/data_road/t
```

1.6.4 Other Information

Follow the README inside the eval folder.

1.7 Eval Results

Results are described below for HED-MAX training set. The warning messages were removed.

```
Starting evaluation ...
Available categories are: ['um_road', 'umm_road', 'uu_road']
Execute evaluation for category um_road ...
Searching for submitted files with prefix: um_road_
MaxF: 97.01
AvgPrec: 87.68
PRE_wp: 94.83
REC_wp: 99.30
FPR_wp: 1.05
FNR_wp: 0.70
('Finished evaluating category: um_road ',)
Execute evaluation for category umm_road ...
Searching for submitted files with prefix: umm_road_
Computing evaluation scores...
MaxF: 97.61
AvgPrec: 89.67
PRE_wp: 96.30
REC_wp: 98.97
FPR_wp: 1.16
FNR_wp: 1.03
('Finished evaluating category: umm_road ',)
Execute evaluation for category uu_road ...
Searching for submitted files with prefix: uu_road_
Computing evaluation scores...
MaxF: 95.42
AvgPrec: 86.48
PRE_wp: 93.77
REC_wp: 97.13
FPR_wp: 1.01
FNR_wp: 2.87
('Finished evaluating category: uu_road ',)
Successfully finished evaluation for 3 categories: ['um_road', 'umm_road', 'uu_road']
```

In []: