

Yolov3 Weightsit Cloudissa

Dataset

Alustavasti hommataan datasetti google driveen seuraavalla sovelluksella:

https://github.com/theAIGuysCode/OIDv4_ToolKit

Ensiksi ladataan git repo komentolinjalla:

```
git clone https://github.com/EscVM/OIDv4_ToolKit.git
```

Sitten ladataan requirementsit:

```
pip3 install -r requirements.txt
```

Kun meidän ympäristö on valmistettu voimme aloittaa kuvien latauksen:

```
python main.py downloader --classes <Sinun haluat luokat> --  
type_csv train --multiclass 1
```

Seuraavaksi varmista, että sinulla on kansiossa classes tekstitiedosto ja se sisältää kaikki valitsemasi luokat. Tarkistuksen jälkeen suorita seuraava linja:

```
python convert_annotations.py
```

Kun koodi on suoritunut, muuta kansion nimeksi obj helpottaakseen jatkoa ja luo google driveen kansio johon siirretään koulutus tiedostot. Siirrä dataset google driveen. Sitten voimme siirtyä Cloudiin suorittamaan koulutusta.


Painojen luonti Cloudissa

Yolov3 painotuksia kouluttamme seuraavassa google colab sivustossa:

https://colab.research.google.com/drive/1Mh2HP_Mfxoao6qNFbhfV3u28tG8jAVGk

Jos olet luomassa painoja ensimmäistä kertaa, käy läpi kaikki vaiheet. Jos colab yhteys on crashannyt, käy läpi vain numeroidut vaiheet.

Aloita collabin käyttö klikkaamalla Open in playground nappia ylhäällä.

 Open in playground

Tarkista, että GPU on käytössä menemällä Edit -> Notebook settings.

Notebook settings

Hardware accelerator

GPU



To get the most out of Colab, avoid using a GPU unless you need one. [Learn more](#)

☐ Omit code cell output when saving this notebook

CANCEL

SAVE

1. Sitten klikataan näitä vuorotellen:

```
[ ] # clone darknet repo
!git clone https://github.com/AlexeyAB/darknet
```

```
[ ] # change makefile to have GPU and OPENCV enabled
%cd darknet
!sed -i 's/OPENCV=0/OPENCV=1/' Makefile
!sed -i 's/GPU=0/GPU=1/' Makefile
!sed -i 's/CUDNN=0/CUDNN=1/' Makefile
```

```
[ ] # verify CUDA
!/usr/local/cuda/bin/nvcc --version
```

```
[ ] # make darknet (build)
!make
```

2. Seuraavaksi suoritetaan tämä:

```
[ ] # define helper functions
def imShow(path):
    import cv2
    import matplotlib.pyplot as plt
    %matplotlib inline

    image = cv2.imread(path)
    height, width = image.shape[:2]
    resized_image = cv2.resize(image, (3*width, 3*height), interpolation = cv2.INTER_CUBIC)

    fig = plt.gcf()
    fig.set_size_inches(18, 10)
    plt.axis("off")
    plt.imshow(cv2.cvtColor(resized_image, cv2.COLOR_BGR2RGB))
    plt.show()

# use this to upload files
def upload():
    from google.colab import files
    uploaded = files.upload()
    for name, data in uploaded.items():
        with open(name, 'wb') as f:
            f.write(data)
            print ('saved file', name)

# use this to download a file
def download(path):
    from google.colab import files
    files.download(path)
```

3. Jonka jälkeen muodostetaan suora yhteys google driveen menemällä linkkiin, jossa kirjaudut omaan google tiliisi. Tämä luo yhteyden sun google driveen avaimen avulla. Avain näkyy ruudulla kun olet kirjautunut tiliisi kopioi siellä oleva avain ja pistä se Enter your authorization code kohtaan.

```
%cd ..  
from google.colab import drive  
drive.mount('/content/gdrive')
```

... /
Go to this URL in a browser: https://accounts.google.com/o/oauth2/authorize?client_id=107717131246005414100&redirect_uri=https://colab.research.google.com/notebooks/authorize_redirect.py&response_type=code
Enter your authorization code:

4. Seuraavalla vaiheella muutetaan driven lähde lyhenettyyn muotoon.

```
[ ] # this creates a symbolic link so that now th
    !ln -s /content/gdrive/My\ Drive/ /mydrive
    !ls /mydrive
```

Sitten lataa yolov3.cfg collabista:

```
[ ] # to download to local machi
    download('cfg/yolov3.cfg')
```

Muuta tätä tiedostoa omien luokkien mukaan rivillä 20 muuta max_batches

2000*(**luokkien määrä**) ja steps= (**sun max_batches** määrä*0.8), (**sun max_batches** määrä *0.9). Classes kohdista muuta kaikki sun luokkien määräksi.

Sitten etsi kaikki yolo kohdat cfg:stä ja sen yläpuolella oleva filter muuta (**sun luokkamäärä + 5**)*3.

Siirrä muokattu cfg tiedosto google driveen. Luo uusi tiedosto: obj.names

obj.names:iin pistä kaikki nimet, jotka löytyvät sun luokissa. Siirrä tiedosto google driveen.

Kopioi seuraavan koodipätkän sisältö ja vie se google driveen:

https://github.com/theAIGuysCode/YoloGenerateTrainingFile/blob/master/generate_train.py/

Sitten klikkaa tämä kohta, tämä luo train.txt tiedoston:

```
[ ] !python generate_train.py
```

Seuraavaksi anna tämän ladata loppuun asti, pjreddie.com sivuston hitauden takia, tämä vaihe vie hieman aikaa.

```
[ ] # upload pretrained convolutional layer weights
!wget http://pjreddie.com/media/files/darknet53.conv.74
```

Sitten luo obj.data tiedosto ja muokkaa sitä sinun tietojen mukaan:

Download

Delete file

Rename file

Copy path

Refresh

classes = **sun luokka määrä**

train = **etsi train.txt tiedosto ja oikea klikkaa copy path**

valid = **ota train kohtan path ja vaiha loppu nimeksi test.txt**

names = **obj.namesin copy path**

backup = /mydrive/**sun kansionnimi**/backup/

Vie obj.data tiedosto google driveen.

5. Klikkaa näitä yksitellen ja jos kansiosi nimi on eri kuin yolov3 muuta näistä yolov3 sen kansion nimeksi.

```
[ ] # this is where my zip is sto
!ls /mydrive/yolov3
```

```
[ ] # copy the .zip file into the
!cp /mydrive/yolov3/obj.zip .
```

```
[ ] # unzip the zip file and its
!unzip ../obj.zip -d data/
```

6. Kopioi tämän vaiheen koodi ja paina CTRL SHIFT i ja pastee se sinne. Tämä varmistaa, että sinua ei timeoutata afkaamisesta.

```
function ClickConnect() {  
  console.log("Working");  
  document.querySelector("colab-toolbar-button#connect").click()  
}  
setInterval(ClickConnect, 60000)
```

Seuraavaksi suorita tämä:

```
[ ] # train your custom detector  
    !./darknet detector train data/obj.data cfg/yolov3_custom.cfg darknet53.conv.74 -dont_show
```

Jos se ei ryhdy suoriutumaan, pistä tiedostoista taas copy path, että tiedostot ovat varmasti oikeat. Minulla muunmuassa näyttää tältä:

```
[ ] # train your custom detector  
    !/content/darknet/darknet detector train /content/darknet/data/obj.data /content/darknet/cfg/yolov3-custom.cfg /content/darknet53.conv.74 -dont_show
```

Nyt pitäisi koulutus olla käynnissä.

7. Jos session crashaa, voit jatkaa seuraaval kohdal kouluksia:

```
[ ] !./darknet detector train data/obj.data cfg/yolov3_custom.cfg /mydrive/yolov3/backup/yolov3_custom_last.weights -dont_show
```