

Instalação de OpenCV e Tensorflow para Python 2 em Linux Mint/Ubuntu

Nota: OpenCV roda melhor em Python2. Em Linux, Tensorflow roda tanto em Python2 como em Python3. Assim, em Linux, vamos rodar OpenCV e Tensorflow em Python2.

Método 1: Usando OpenCV que vem no Cekeikon

1) Instale Cekeikon5.6. Ative-o.

```
$ source ~/cekeikon5/bin/ativa_cekcpu  
(CekCPU)$
```

2) Instale Tensorflow e Python2 num ambiente virtual (para não interferir com outros ambientes de Python).

```
mkdir ~/tensorflow # Algum diretorio de trabalho  
cd ~/tensorflow  
mkdir cpu # Nome de alguma versao de tensorflow  
sudo apt install virtualenv  
virtualenv --system-site-packages cpu
```

```
source ~/tensorflow/cpu/bin/activate #Ativa ambiente cpu do tensorflow
```

Nota: para sair do ambiente virtual: deactivate

```
(cpu) (CekCPU)$ pip install --upgrade pip
```

```
(cpu) (CekCPU)$ pip install -U tensorflow
```

3) Copie

```
~/cekeikon5/opencv3cpu/lib/python2.7/dist-packages/cv2.so
```

para

```
~/tensorflow/cpu/lib/python2.7/site-packages:
```

```
sudo cp ~/cekeikon5/opencv3cpu/lib/python2.7/dist-packages/cv2.so ~/tensorflow/cpu/lib/python2.7/site-packages
```

Nota: Se quiser usar OpenCV2 dentro do Python, escreva opencv2cpu.

4) Teste instalação de Tensorflow/Keras e OpenCV em Python2:

```
(cpu) (CekCPU)$ python -c "import tensorflow as tf; print(tf.__version__)"
```

```
1.12.0
```

```
(cpu) (CekCPU)$ python -c "import tensorflow.keras as keras; print(keras.__version__)"
```

```
2.1.6-tf
```

```
(cpu) (CekCPU)hae@Royale ~ $ python -c "import cv2; print(cv2.__version__)"
```

```
3.4.3
```

Nota: Se o seu computador tiver GPU e bibliotecas instaladas, pode criar um ambiente gpu (em vez de cpu) e instalar a versão gpu do tensorflow

```
diretorio$ pip install -U tensorflow-gpu
```

Pronto!

Método 2: Usando OpenCV independente do Cekeikon

1) Vamos supor que no computador não está instalado Cekeikon. Se Cekeikon estiver instalado, deve desabilitar o uso de OpenCV do Cekeikon pelo Python. Abra arquivo `~/cekeikon5/bin/ativa_cekcpu` (ou `ativa_cekgpu` se você estiver usando a versão para GPU) e comente as linhas:

```
#export PYTHONPATH=$CEK5_PATH/opencv2$CEK5_MODE/lib/python2.7/dist-packages:$PYTHONPATH
#export PYTHONPATH=$CEK5_PATH/opencv3$CEK5_MODE/lib/python2.7/dist-packages:$PYTHONPATH
```

Comente as duas linhas, colocando um "#" na frente.

2) Instale Tensorflow e Python2 num ambiente virtual (para não interferir com outros ambientes de Python).

```
mkdir ~/tensorflow # Algum diretorio de trabalho
cd ~/tensorflow
mkdir cpu # Nome de alguma versao de tensorflow
sudo apt install virtualenv
virtualenv --system-site-packages cpu

source ~/tensorflow/cpu/bin/activate #Ativa ambiente cpu do tensorflow
Nota: para sair do ambiente virtual: deactivate
```

```
(cpu) (CekCPU)$ pip install --upgrade pip
(cpu) (CekCPU)$ pip install -U tensorflow
```

3) Instale OpenCV dentro do ambiente virtual

```
(cpu) (CekCPU)$ pip install opencv-python
```

4) Teste instalação de Tensorflow/Keras e OpenCV em Python2:

```
(cpu) (CekCPU)$ python -c "import tensorflow as tf; print(tf.__version__)"
1.12.0
(cpu) (CekCPU)$ python -c "import tensorflow.keras as keras; print(keras.__version__)"
2.1.6-tf
(cpu) (CekCPU)hae@Royale ~ $ python -c "import cv2; print(cv2.__version__)"
3.4.3
```

Nota: Se o seu computador tiver GPU e bibliotecas instaladas, pode criar um ambiente gpu (em vez de cpu) e instalar a versão gpu do tensorflow

```
diretorio$ pip install -U tensorflow-gpu
```

Pronto!