

C++/OpenCV

Última revisão: 05/03/2024 - Hae Yong Kim

Este documento descreve como compilar e executar programas em C++/OpenCV em Linux (Ubuntu/Mint) e Raspberry, sem a biblioteca Cekeikon.

Este procedimento não foi suficientemente testado. Se você quiser fazer uma instalação mais testada, instale o pacote Cekeikon. Você também fazer instalação seguindo o site do OpenCV:

https://docs.opencv.org/4.2.0/d7/d9f/tutorial_linux_install.html

1. Em Linux Mint/Ubuntu ou Windows com WSL/VirtualBox:

Em Windows, você pode instalar primeiro WSL (Windows Subsystem for Linux) ou instalar Linux dentro de VirtualBox e seguir o mesmo procedimento.

1) Se você já instalou Cekeikon, verifique se o arquivo ~/.bashrc está ativando a biblioteca Cekeikon. Se estiver, apague ou comente essa linha com #:

```
export PATH=".:$HOME/bin:$PATH"
#source ~/cekeikon5/bin/ativa_cekcpu
```

2) Atualize o seu sistema Linux:

- `sudo apt-get update`

3) Instale os programas:

- `sudo apt-get install build-essential`
- `sudo apt-get install cmake git libgtk2.0-dev pkg-config libavcodec-dev libavformat-dev libswscale-dev`
- `sudo apt-get install libopencv-dev`
- (opcional) `sudo apt-get install libtbb2 libtbb-dev libjpeg-dev libpng-dev libtiff-dev`
- `sudo apt-get autoremove`

4) Copie o seguinte script para um editor de texto e salve-o como **compila.sh** em algum diretório listado no PATH:

```
1 set -v
2 g++ -std=gnu++14 $1.cpp -o $1 -fmax-errors=2 `pkg-config opencv4 --libs --cflags` -O3 -s $2 $3 $4 $5 $6 $7
3 set +v
```

Depois, torne-o executável com o comando:

```
$ chmod +x compila.sh
```

Nota: É possível descobrir os diretórios do PATH com o comando:

```
$ echo $PATH
```

Nota: Se no seu sistema estiver instalado OpenCV2 ou OpenCV3, mude a segunda linha de "opencv4" para "opencv".

5) Copie o seguinte programa para um editor de texto e salve-o como ***hello_opencv.cpp*** em algum diretório de trabalho (pode ser diferente do diretório onde está compila.sh):

```
#include <opencv2/opencv.hpp>
using namespace std;
using namespace cv;

int main()
{ Mat_<Vec3b> a(300,300, Vec3b(255,255,0));
  for (int l=100; l<200; l++)
    for (int c=100; c<200; c++)
      a(l,c)=Vec3b(0,255,255);
  namedWindow("janela");
  imshow("janela",a);
  waitKey();
}
```

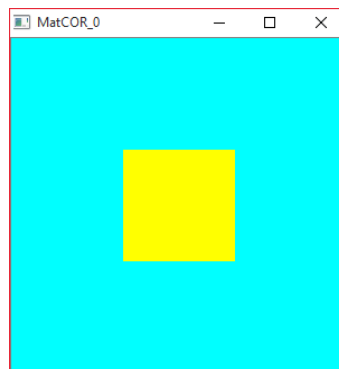
6) Compile-o:

```
$ compila.sh hello_opencv
```

7) Execute-o:

```
$ ./hello_opencv
```

Deve aparecer a janela:



Pronto!

2. No Raspberry Pi:

1) Se Cekeikon estiver instalado no seu Raspberry, edite o arquivo `~/.bashrc` apagando ou comentando essa linha com #:

```
export PATH=".:$HOME/bin:$PATH"
#source ~/cekeikon5/bin/inst-cek5-raspberry.sh
```

2) Atualize o seu sistema:

- `sudo apt-get update`

3) Instale os programas:

- `sudo apt-get install build-essential`
- `sudo apt-get install cmake git libgtk2.0-dev pkg-config libavcodec-dev libavformat-dev libswscale-dev`
- `sudo apt-get install libopencv-dev`
- (opcional) `sudo apt-get install libtbb2 libtbb-dev libjpeg-dev libpng-dev libtiff-dev`
- `sudo apt-get autoremove`

4) Copie o seguinte script para um editor de texto e salve-o como ***compila.sh*** em algum diretório listado no PATH (por exemplo, no diretório `~/bin`):

```
1 set -v
2 g++ -std=gnu++14 $1.cpp -o $1 -fmax-errors=2 `pkg-config opencv --libs --cflags` -O3 -s $2 $3 $4 $5 $6 $7
3 set +v
```

Depois, torne-o executável com o comando:

```
$ chmod +x compila.sh
```

Nota: É possível descobrir os diretórios do PATH com o comando:

```
$ echo $PATH
```

Nota: Se no seu sistema estiver instalado OpenCV4, mude a segunda linha de “opencv” para “opencv4”.

5) Copie o seguinte programa para um editor de texto e salve-o como ***hello_opencv.cpp*** em algum diretório de trabalho (pode ser diferente do diretório onde está `compila.sh`):

```
#include <opencv2/opencv.hpp>
using namespace std;
using namespace cv;

int main()
{ Mat_<Vec3b> a(300,300, Vec3b(255,255,0));
  for (int l=100; l<200; l++)
    for (int c=100; c<200; c++)
      a(l,c)=Vec3b(0,255,255);
  namedWindow("janela");
  imshow("janela",a);
  waitKey();
}
```

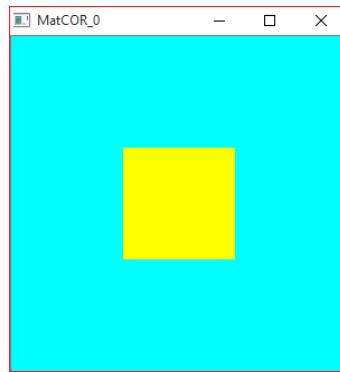
6) Compile-o:

```
$ compila.sh hello_opencv
```

7) Execute-o:

```
$ ./hello_opencv
```

Deve aparecer a janela:



Pronto!