

# Robô seguidor de linha utilizando Raspberry Pi, Camera USB, Python e OpenCV

**Resumo:** Projeto de um programa em Python com OpenCV no Raspberry Pi para detectar caminhos.

**Python:** Python foi escolhida como linguagem por dois motivos: O tempo de desenvolvimento nela é menor e que seu código é de fácil leitura.

**OpenCV:** OpenCV é uma biblioteca de visão computacional e foi escolhida para que processe as imagens vinda da câmera.

## Classes, Atributos e Métodos:

### Program:

Classe responsável por gerenciar as demais e executar o programa, é a primeira classe a ser executada.

Seus atributos são:

- conf - Instância da classe Config;
- camera - Instância da classe Camera;
- imEfc - Instância da classe ImageEffects;
- lineD - Instância da classe LineDetect;
- gC - Instância da classe GpioController;

Seus métodos são:

- \_\_init\_\_ - Inicializa a instância;
- Execute - Onde todo o programa roda.

### Config:

Classe responsável por gerenciar as configurações nela salvas.

Seus atributos são:

- conf - Array com todas as configurações:
  - self.conf['cam'] - Qual camera o programa usará
  - self.conf['camW'] - Qual a largura que deverá ser usada
  - self.conf['camH'] - Qual a altura que deverá ser usada
  - self.conf['imgX'] - Utilizada no corte da imagem
  - self.conf['imgY'] - Utilizada no corte da imagem
  - self.conf['imgW'] - Utilizada no corte da imagem
  - self.conf['imgH'] - Utilizada no corte da imagem
  - self.conf['Esq'] - Pino da esquerda
  - self.conf['Dir'] - Pino da direita
  - self.conf['GpioMode'] - Modo da Gpio
  - self.conf['freq'] - Frequência que deverá ser usada

Seus métodos são:

- `__init__` - Inicializa a instância e as configurações;
- `getConfig` - Retorna a configuração desejada.

## Camera:

Classe responsável por capturar imagens da camera.

Seus atributos são:

- `video_capture` - Instância da classe `VideoCapture` da biblioteca `OpenCV`.

Seus métodos são:

- `__init__` - Inicializa a instância com as configurações;
- `captureFrame` - Captura e retorna a imagem da Camera.

## ImageEffects:

Classe responsável por aplicar efeitos na imagem.

Não possui atributos.

Seus métodos são:

- `__init__` - Inicializa a instância com as configurações;
- `work` - Transforma a imagem em imagem transparente, borra ela (suaviza retirando ruídos) e transforma em uma imagem binária (total branco ou total preto).

## LineDetect:

Classe responsável por detectar o contorno da linha.

Não possui atributos.

Seus métodos são:

- `__init__` - Inicializa a instância com as configurações;
- `findContour` - Encontra e retorna o contorno da linha;
- `getMov` - Calcula e retorna o centro do contorno.

## GpioController:

Classe responsável por controlar a GPio do Raspberry Pi.

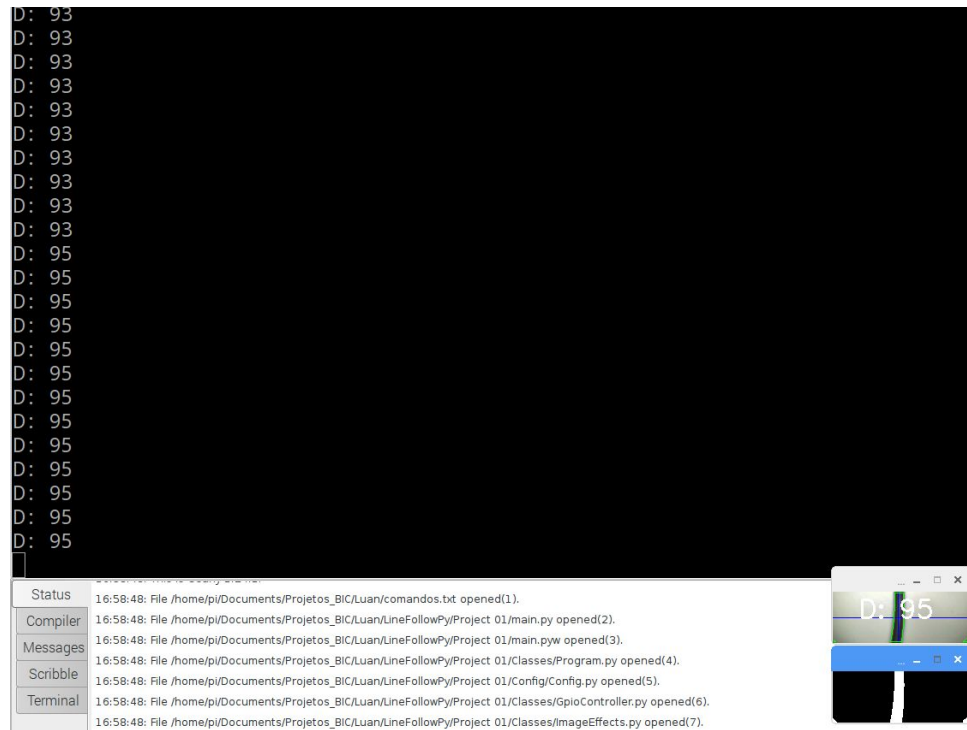
Seus atributos são:

- `pwmEsq` - Instância da classe `PWM` da biblioteca `GPio`; Gerencia o pino esquerdo;
- `pwmDir` - Instância da classe `PWM` da biblioteca `GPio`; Gerencia o pino direito.

Seus métodos são:

- `__init__` - Inicializa a instância com as configurações;
- `setDir` - A partir do meio do contorno calculado determina com que força e direção cada pino deve estar.

## Imagens:



## Instalação da biblioteca OpenCV para python no Raspberry Pi:

Comandos no terminal:

```
sudo rpi-update
sudo reboot
sudo apt-get install build-essential git cmake pkg-config
sudo apt-get install libjpeg-dev libtiff5-dev libjasper-dev
libpng12-dev
sudo apt-get install libavcodec-dev libavformat-dev
libswscale-dev libv4l-dev
sudo apt-get install libxvidcore-dev libx264-dev
sudo apt-get install libgtk2.0-dev
sudo apt-get install libatlas-base-dev gfortran
cd ~
git clone https://github.com/Itseez/opencv.git
cd opencv
git checkout 3.1.0
cd ~
git clone https://github.com/Itseez/opencv_contrib.git
cd opencv_contrib
git checkout 3.1.0
sudo apt-get install python2.7-dev
wget https://bootstrap.pypa.io/get-pip.py
sudo python get-pip.py
pip install numpy
cd ~/opencv
mkdir build
cd build
cmake -D CMAKE_BUILD_TYPE=RELEASE \
      -D CMAKE_INSTALL_PREFIX=/usr/local \
      -D INSTALL_C_EXAMPLES=OFF \
      -D INSTALL_PYTHON_EXAMPLES=ON \
      -D
      OPENCV_EXTRA_MODULES_PATH=~/opencv_contrib/modules \
      -D BUILD_EXAMPLES=ON ..
make -j4
sudo make install
sudo ldconfig
```