



Desenvolvimento de algoritmo de imagem baseado em python para detecção de sonolência utilizando raspberry pi

Vitor A. Dias*, Pedro H. de Oliveira, Gabriel H. Vicente, Gabriel S. da Silva, César H. C. Quiroz, Alan F. P. Tavares, Paulo R. G. Kurka

Resumo

O presente projeto trata sobre a utilização de visão computacional, processamento de imagem e sistemas embarcados. O principal objetivo foi desenvolver um algoritmo baseado em processamento de imagem e python para a detecção de sonolência em motoristas. O projeto foi dividido em etapas para a compreensão de assuntos mais complexos, pois torna-se fundamental o conhecimento em matemática, lógica de programação C/C++, Python, Dlib e OpenCV para o seu desenvolvimento. Os resultados iniciais demonstraram uma acurácia de 85%, pois o sistema depende de boa iluminação. Em horários noturnos, torna-se difícil a identificação dos olhos e conseqüentemente a análise do estado de sonolência – para tal solução poderia ser usada uma câmera infravermelha e desta maneira validar o sistema em turnos diferentes.

Palavras-chave:

Processamento de imagem, Python, OpenCV.

Introdução

A visão computacional está presente na atualidade como a tecnologia que mais cresce, abrangendo um mercado em crescimento exponencial em diversas áreas do conhecimento que possam ser aplicadas. O objetivo básico é extrair informações úteis das imagens de acordo com a aplicação¹. O objetivo principal do projeto foi desenvolver um algoritmo em python para detectar a sonolência em usuários. Foram utilizadas bibliotecas como OpenCV e Dlib para detectar os marcos faciais ou landmarks na face do usuário. O sistema foi implementado no raspberry pi 3, onde é possível detectar quantas piscadas são realizadas e se caso o usuário permaneça com os olhos fechados durante 4 segundos, é considerado estado de sonolência. Após esta mudança de estado, o usuário é alertado com uma sirene.

Resultados e Discussão

A obtenção dos resultados iniciais partiram das aulas de lógica de programação, linguagem de programação em python, processamento de imagem e linux. O algoritmo foi desenvolvido com base no script do autor Adrian Rosebrock². Para a implementação do algoritmo foram necessárias adicionar funções para detectar e desenhar a borda do olho. Isto foi implementado justamente para conseguir obter a mudança no estado de sonolência do usuário. A Figura 1, demonstra a região do olho onde foram traçadas linhas para a detecção de piscada. Foi utilizada uma relação de aspecto do olho (EAR), onde a Equação 1 é descrita a seguir.

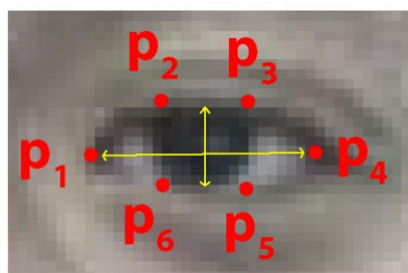
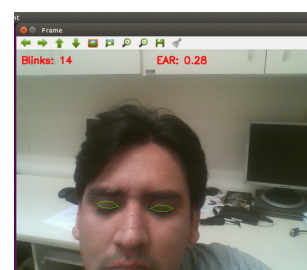
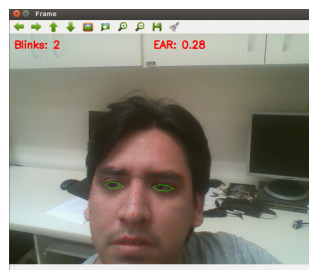


Figura 1. Relação de aspecto do olho

$$EAR = \frac{(p_2 - p_6) + (p_3 - p_5)}{2(p_1 - p_4)}$$

onde p, representa as localizações 2D dos marcadores faciais. Quando os pontos se tornam praticamente alinhados entre si, o EAR diminui para aproximadamente 0.1, ou seja, isto significa que ocorreu um “piscar” do usuário.

A seguir são demonstrados o resultados da implementação integrada. Os marcadores faciais são demonstrados, bem como a detecção do olho.



Conclusões

O algoritmo desenvolvido obteve uma robustez de aproximadamente 85% em relação as medidas obtidas. Iluminação é um fator que influencia na acurácia dos dados, no entanto, em casos noturnos, pode-se utilizar uma câmera infravermelha e avaliar o algoritmo proposto. O algoritmo está disponibilizado no bitbucket (<https://bitbucket.org/VitinhoBarata/pibic2019/src/master/>)

Agradecimentos

Nosso singelo agradecimentos à Unicamp e Prof. Dr. Paulo Roberto Gardel Kurka.

¹ BASTOS, Vinicius B. et al. Monocular visual odometry for robotic wheelchair in a virtual environment. In: **2018 Annual IEEE International Systems Conference (SysCon)**. IEEE, 2018. p. 1-8.

² Adrian Rosebrock: <https://www.pyimagesearch.com/2017/04/17/real-time-facial-landmark-detection-opencv-python-dlib/>