

UNIVERSIDADE PRESBITERIANA MACKENZIE
FACULDADE DE COMPUTAÇÃO E INFORMÁTICA
COMPUTAÇÃO VISUAL - 07G

Aluno: Daniel Faia Monteiro da Silva
TIA: 31932029

Operadores de gradiente Roberts, Prewitt e Sobel

A detecção de bordas é um dos processos mais utilizados quando se fala de computação visual ou, mais especificamente, de processamento de imagens. A motivação disso é que, de acordo com J.R. Parker (2011), as arestas formam o contorno de um objeto e objetos são assuntos de interesse em análise de imagens e no campo da visão computacional.

Segundo Patrícia Maturana (2010), para detectar uma borda em uma imagem é necessário aplicar um filtro por derivada utilizando máscaras de convolução, que também são conhecidas como operadores 2x2 ou 3x3.

Alguns exemplos dessas máscaras são os operadores de gradiente Roberts, Prewitt e Sobel, que é o assunto tratado nesta pesquisa.

Sérgio Brito (1998) afirma que existem dois operadores gradiente fundamentais de Primeira Ordem (ou detectores de borda diferenciadores de primeira ordem). Um método envolve a geração de gradientes em duas direções ortogonais na imagem enquanto o outro método utiliza um conjunto de derivadas direcionais.

Existem várias bibliotecas que auxiliam no processamento de imagens, mas nesta pesquisa será utilizada a scikit-image.

Primeiro importa-se as bibliotecas necessárias:

```
import numpy as np
import matplotlib.pyplot as plt

from skimage.io import imread
from skimage.filters import roberts, sobel, scharr, prewitt
```

Em seguida, fazemos a leitura de uma imagem e, então, aplicamos os operadores de Roberts, Sobe, Scharr e Prewitt:

```
img = imread("input/camera_01.jpg", as_gray=True) # carregando imagem
op_roberts = roberts(img)
op_sobel = sobel(img)
op_scharr = scharr(img)
op_prewitt = prewitt(img)
```

Como é mostrado na imagem, a própria biblioteca scikit-imagem possui os métodos de aplicação dos operadores tratados nesta pesquisa, facilitando o trabalho.

Por fim, plotamos as imagens para visualizar os resultados obtidos após a aplicação dos operadores:

```
fig, axes = plt.subplots(nrows=2, ncols=2, sharex=True, sharey=True, figsize=(12, 12))
ax = axes.ravel()

ax[0].imshow(op_roberts, cmap=plt.cm.gray)
ax[0].set_title('Operador de Roberts')

ax[1].imshow(op_sobel, cmap=plt.cm.gray)
ax[1].set_title('Operador de Sobel')

ax[2].imshow(op_scharr, cmap=plt.cm.gray)
ax[2].set_title('Operador de Scharr')

ax[3].imshow(op_prewitt, cmap=plt.cm.gray)
ax[3].set_title('Operador de Prewitt')

for a in ax:
    a.axis('off')

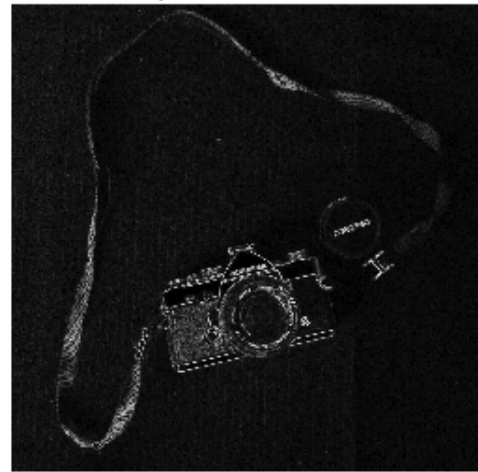
plt.tight_layout()
plt.show()
```

Resultados obtidos:

Operador de Roberts



Operador de Sobel



Operador de Scharr



Operador de Prewitt



REFERÊNCIAS

PARKER, J.R. **Algorithms for Image Processing and Computer Vision**. Wiley Publishing, 2011.

MATURANA, Patrícia. **ALGORITMOS DE DETECÇÃO DE BORDAS IMPLEMENTADOS EM FPGA**. FACULDADE DE ENGENHARIA DE ILHA SOLTEIRA, 2010.

BRITO, Sérgio. **Relatorio Final de Iniciação Científica Sistemas de Processamento Digital de Imagens para Fins Didático/Científico - Estudo Seleção e Implementação de Algoritmos de Segmentação**. Universidade Federal da Paraíba, 1998.

SOUZA, Richardson. **Processamento de Imagem em Python**, 2018. Disponível em: [GitHub - richardson-souza/processamento-de-imagem: Exemplos de processamento de imagem em python \(scikit-image\)](https://github.com/richardson-souza/processamento-de-imagem-exemplos-de-processamento-de-imagem-em-python-scikit-image)>.