

- Tarefa 1

A aplicação foi escrita em C++, utilizando o OpenCV versão 4.1.2. O código fonte completo pode ser encontrado na pasta src. Anexado a este documento encontra-se um link de acesso a um vídeo gravado usando as funções do OpenCV, no qual há uma demonstração de vários resultados obtidos na execução das tarefas abaixo. Também anexado a este documento encontra-se uma tabela com os comandos de utilização da aplicação.

- Tarefa 2

A aplicação o comando GaussianBlur depende de duas variáveis: dimensão do Kernel e desvio padrão na direção X (os outros parâmetros são usados com o valor default). A dimensão do Kernel é definida pelo usuário por meio de uma trackbar e o desvio padrão é arbitrariamente colocado em zero. Vale notar que estamos usando um kernel quadrado, portanto a escolha de valores pares para o lado do kernel causa problemas. Tratamos esse caso acrescentando 1 aos valores pares dados como entrada.

O resultado da aplicação do GaussianBlur pode ser encontrado no link anexado a este documento.

- Tarefa 3

Dois detalhes chamaram atenção na realização desta tarefa.

- 1) Para chamarmos o comando Canny precisamos primeiro converter a imagem para tons de cinza.
- 2) A detecção de arestas de Canny é extremamente sensível ao ruído. Para obtermos resultados razoáveis precisamos de antemão suavizar a imagem, aplicando GaussianBlur, por exemplo.

O resultado da aplicação do detector de arestas de Canny pode ser encontrado no link anexado a este documento.

- Tarefa 4

A estimativa do gradiente da imagem usando o comando Sobel possui duas variáveis importantes:

- 1) Ordem da derivada em relação ao eixo X
- 2) Ordem da derivada em relação ao eixo Y

A implementação disponibilizada tem como default o uso da derivada de segunda ordem, tanto para X quanto para Y. No entanto, tais valores podem ser modificados facilmente trocando o valor dos macros DX e DY.

O resultado da estimativa do gradiente usando Sobel pode ser encontrado no link anexado a este documento.

- Tarefa 5

O comando `convertTo` aplica uma transformação linear pontual na imagem. Para controlarmos os valores de gain (contraste) e bias (brilho) usamos duas trackbars. Na chamada do comando, foram usados os seguintes valores:

- 1) Cálculo do negativo:
 - Gain = -1
 - Bias = 255
- 2) Ajuste de contraste:
 - Gain = valor da trackbar
 - Bias = 0
- 3) Ajusto do brilho:
 - Gain = 1
 - Bias = valor da trackbar

O resultado da aplicação do cálculo de negativo e dos ajustes de contraste e brilho pode ser encontrado no link anexado a este documento.

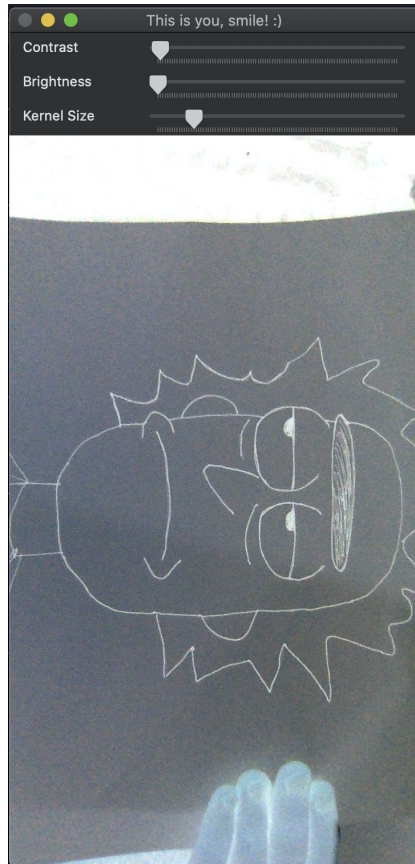
- Tarefa 6 e 7

A conversão para tons de cinza e o redimensionamento do frame foram as atividades mais facilmente realizadas, pois a implementação é direta. Para a conversão para tons de cinza usou-se a função `cvtColor(src, dst, COLOR_BGR2GRAY)`. Para o redimensionamento usou-se a função `resize(src, dst, Size(newRows, newCols))`, onde `Size(newRows, newCols)` representa as dimensões da nova imagem. O resultado da aplicação do redimensionamento pode ser visto abaixo. O resultado da conversão para tons de cinza pode ser encontrado no link anexado a este documento.



- Tarefa 8

A rotação foi certamente a operação mais complicada. Por meio de uma pesquisa na web encontrou-se um método para a rotação de 90 graus no sentido horário de uma imagem I que consiste em primeiramente calcular a matriz transposta de I e depois aplicar um espelhamento em relação ao eixo vertical no resultado do cálculo da transposta de I . O resultado de uma rotação de 90 graus pode ser visto abaixo.



- Tarefa 9

Diferentemente da rotação, o espelhamento teve implementação direta usando o comando flip. Este comando possui uma flag indicadora do eixo de espelhamento. Para espelhamento em relação ao eixo horizontal, usa-se a flag 0. Para espelhamento em relação ao eixo vertical, usa-se a flag 1.

O resultado da aplicação de espelhamentos pode ser encontrado no link anexado a este documento.

- Tarefa 10

A realização dessa tarefa pode ser visualizada seguindo o link disponibilizado em anexo.

ANEXO:

1) Link para a visualização do vídeo:

<https://www.youtube.com/watch?v=ozO8DGowdEo>

2) Tabela de comandos da aplicação:

Caractere	Comando
'1'	Gaussian Blur
'2'	Canny Edge Detection
'3'	Sobel Gradient Estimative
'4'	Negative
'5'	90 Degree Clockwise Rotation
'6'	Horizontal Mirroring
'7'	Vertical Mirroring
'8'	Conversion to Grayscale
'9'	Resize Image
'0'	Reset All Changes
'a'	Start Recording
'b'	Stop Video Recording