UNIVERSIDADE FEDERAL DO
RIO GRANDE DO SUL
UFRGS

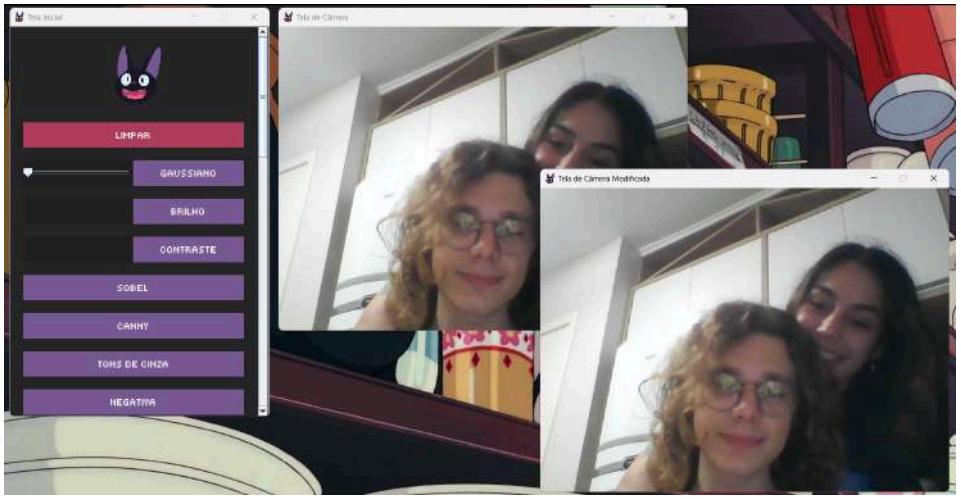
RELATÓRIO DE FUNDAMENTOS DE PROCESSAMENTO DE IMAGENS - TRABALHO 3

RICARDO ZANINI DE COSTA

Porto Alegre, Novembro de 2024

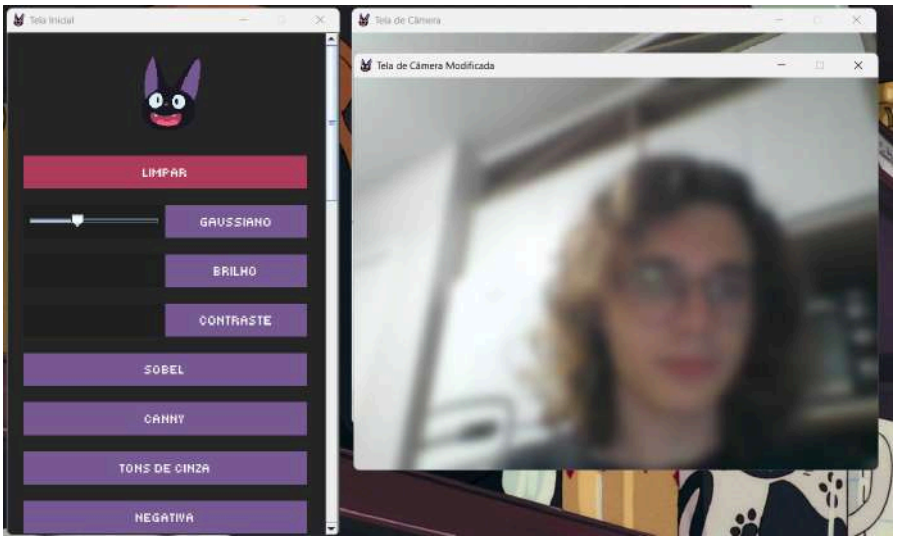
1. EXERCÍCIO 01 - Configurar Ambiente

A configuração do ambiente em Java utilizando OpenCV foi um pouco complicada, principalmente a configuração do arquivo “dll” e o processo de entendimento de configuração da câmera utilizando a classe VideoCapture.



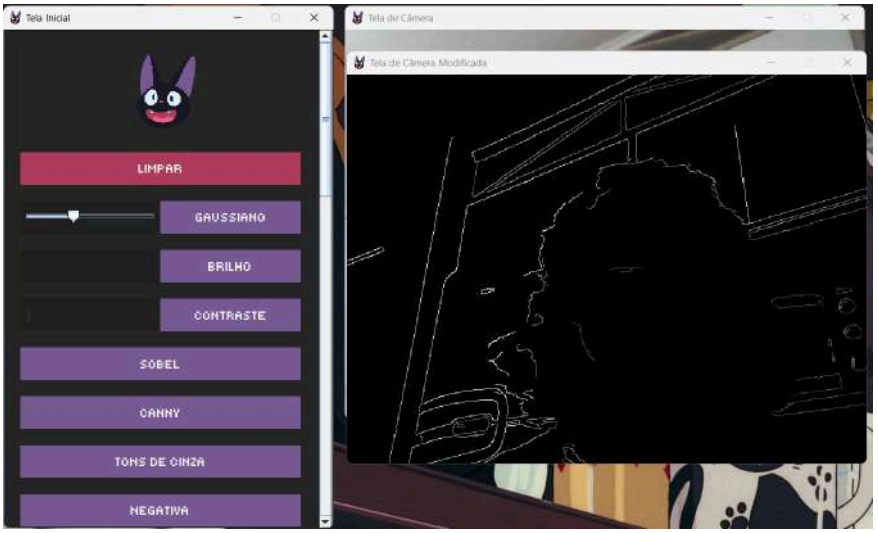
2. EXERCÍCIO 02 - Gaussian Blur

A implementação desse filtro utilizou a função GaussianBlur do OpenCV, sendo enviado para ela o frame atual da imagem bem como as dimensões do filtro de blur, ditados por uma “trackbar” na tela inicial. Essa trackbar não pode gerar valores pares, visto que isso faria com que a matriz do filtro não funcionasse. Portanto, sempre que um valor par acaba sendo selecionado na trackbar, esse é diminuído em 1 produzindo um valor ímpar.



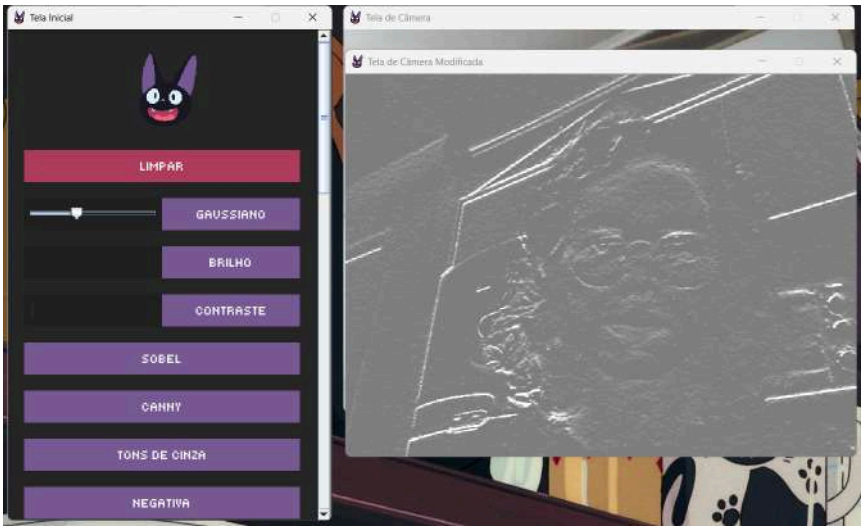
3. EXERCÍCIO 03 - Canny

O filtro Canny foi implementado com o auxílio de outras funções, sendo a imagem passada para a escala de cinza e após isso é aplicado um filtro blur simples, para a obtenção de um melhor resultado do comando canny.



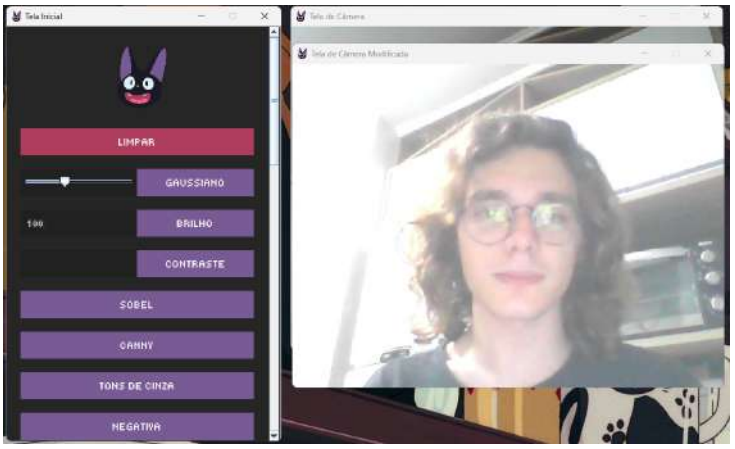
4. EXERCÍCIO 04 - Sobel

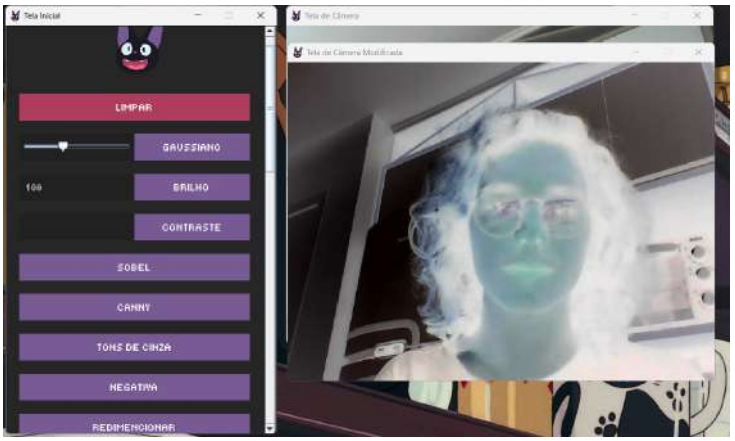
Algo muito semelhante foi feito para o caso deste efeito, em que foi feita uma conversão para uma escala de cinza antes da aplicação do sobel, e após a aplicação foi somado o valor 127 ao valor dos pixels, para que a impressão de um relevo fosse passada.

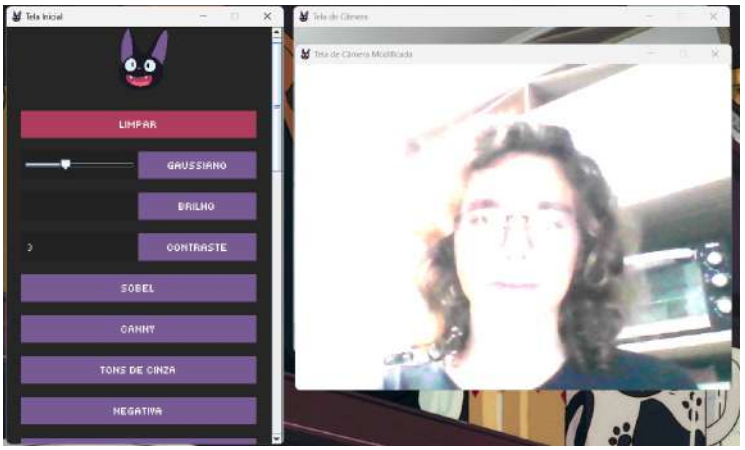


5. EXERCÍCIO 05 - Brilho, Negativo e Contraste

Para a aplicação dos filtros de brilho e contraste foi necessário a adição de dois novos campos na tela inicial para a obtenção dos valores utilizados no cálculo desses filtros. Para a aplicação do filtro negativo o que foi feito foi a multiplicação do valor do pixel por -1 seguida da soma com 255, fazendo o negativo.

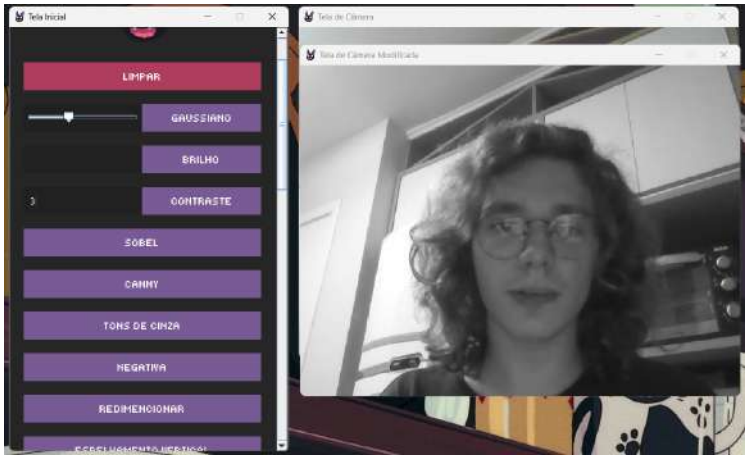






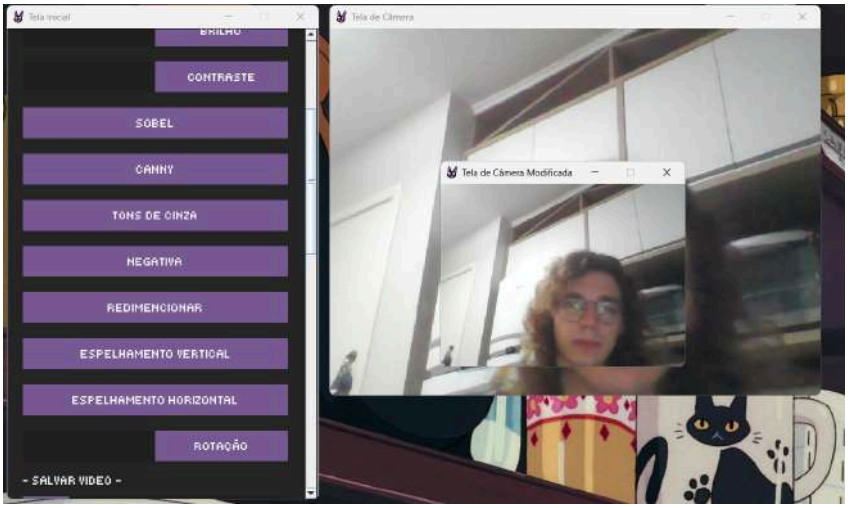
6. EXERCÍCIO 06 - Grayscale

Essa função foi facilmente implementada por meio do comando cvtColor presente no openCV, convertendo uma imagem de um domínio de 3 cores para um domínio de uma cor, no caso tons de cinza.



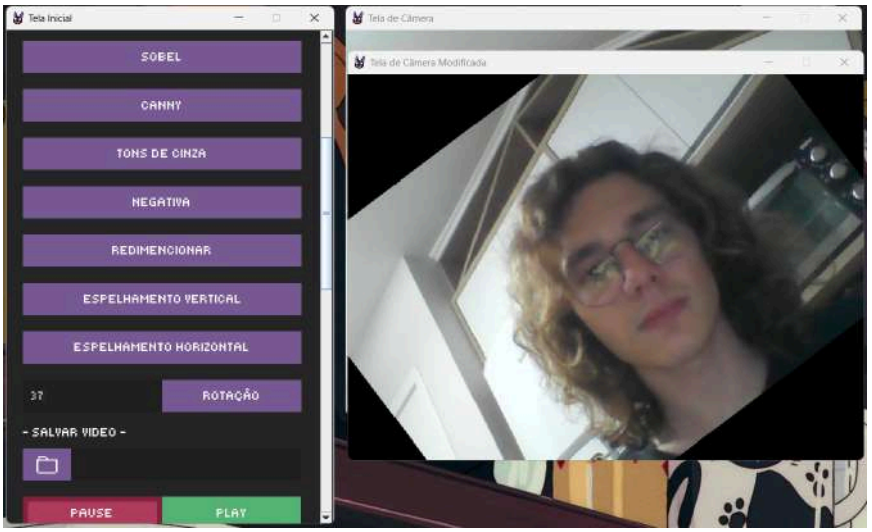
7. EXERCÍCIO 07 - Redimensionamento

O redimensionamento foi feito por meio do comando resize do opencv, que permite redimensionar uma imagem da maneira desejada contanto que se envie as novas medidas. Para esse caso as medidas enviadas foram metade da altura e metade da largura da imagem, assim cortando cada dimensão por um fator 2.



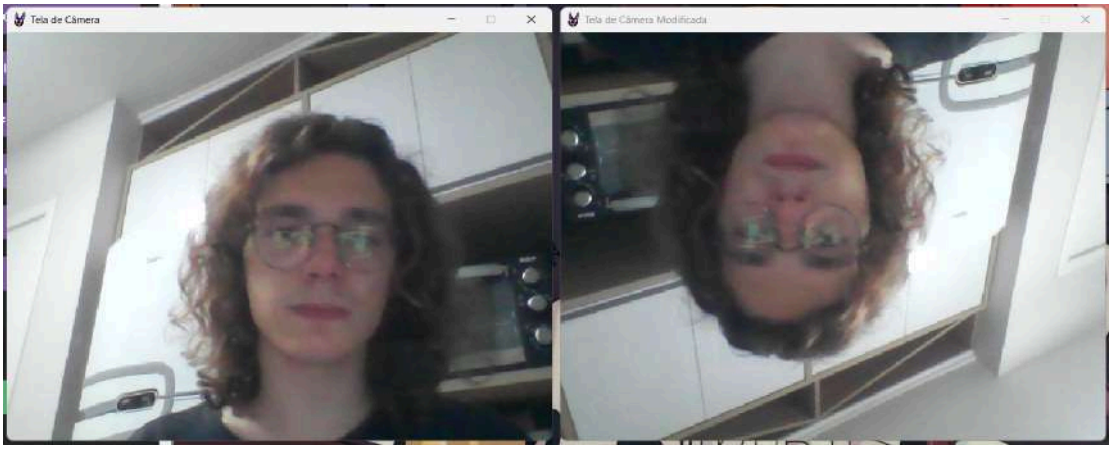
8. EXERCÍCIO 08 - Rotação

Apliquei essa função de modo que a rotação seja possível para todos os ângulos, porém essa configuração me impediu de fazer um correto redimensionamento da imagem.



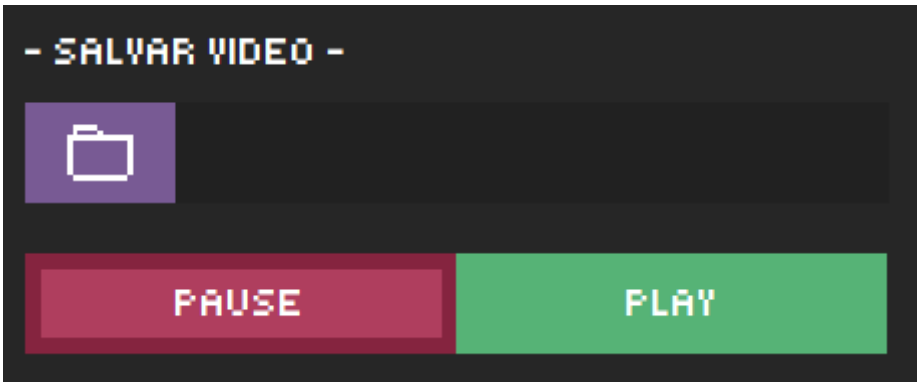
9. EXERCÍCIO 09 - Espelhamento

Foi facilmente implementado pelo comando flip, que dependendo de seus parâmetros pode girar a imagem no eixo vertical ou horizontal (flipcode 0 para vertical e flipcode 1 para horizontal).



10. EXERCÍCIO 10 - Gravação do Vídeo

A gravação de vídeo foi de longe a etapa mais complexa do trabalho, devido a uma série de bugs que ocorreram durante seu desenvolvimento, os quais pude compreender e corrigir ao longo do projeto. Sua implementação basicamente consiste da inicialização de uma classe VideoWriter e da sua escrita por meio do método write, com sua finalização por meio do método release.



11. OBSERVAÇÃO

Para utilizar o programa é necessária uma correta configuração do arquivo “dll” da biblioteca OpenCV, caso contrário o programa produzirá erro ao ser compilado. Optei por tornar todas as operações no programa concatenáveis, portanto caso deseje-se aplicar apenas uma operação sobre a imagem o botão “limpar” deve ser pressionado para remover todas operações e então aplicar a função de sua preferência.