

Relatório Trabalho 2 - FPI

Aluno: Vilmar Dorneles Aprato Neto

Dificuldades:

As únicas funções que não foram implementadas a tempo foram as de zoom-in , zoom-out e o histograma com equalização. Houveram dificuldades, pois o aluno não anotou o horário de entrega e tinha pensado que era para terça , dia 20.

Além disso, devido a desatenção do aluno aqui presente, não foram implementadas na GUI devido ao pouco tempo que ele percebeu que devia fazer tudo.

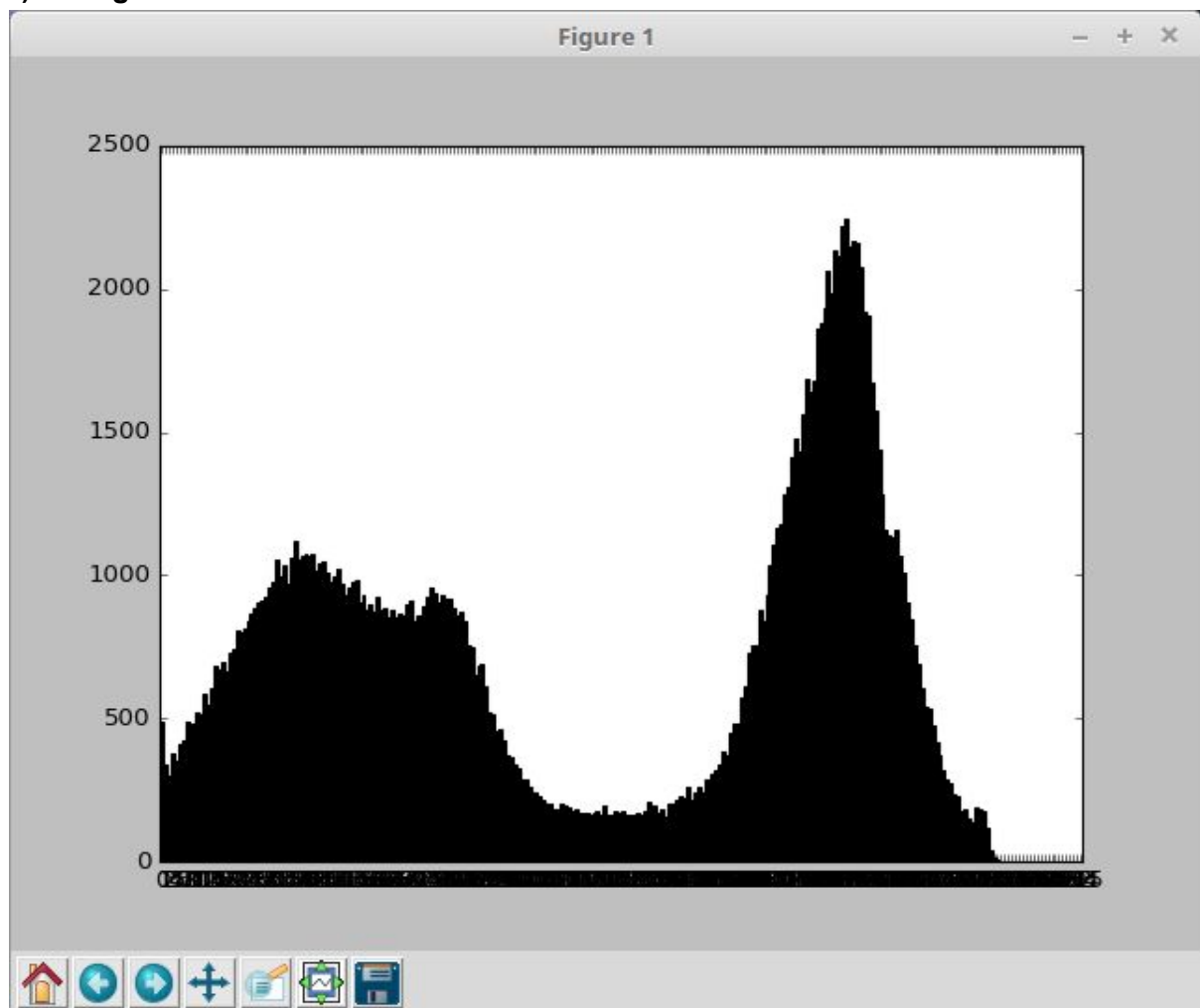
Código:

O código se encontra no link <https://github.com/Dkmister/Trabalho_1-FPI>

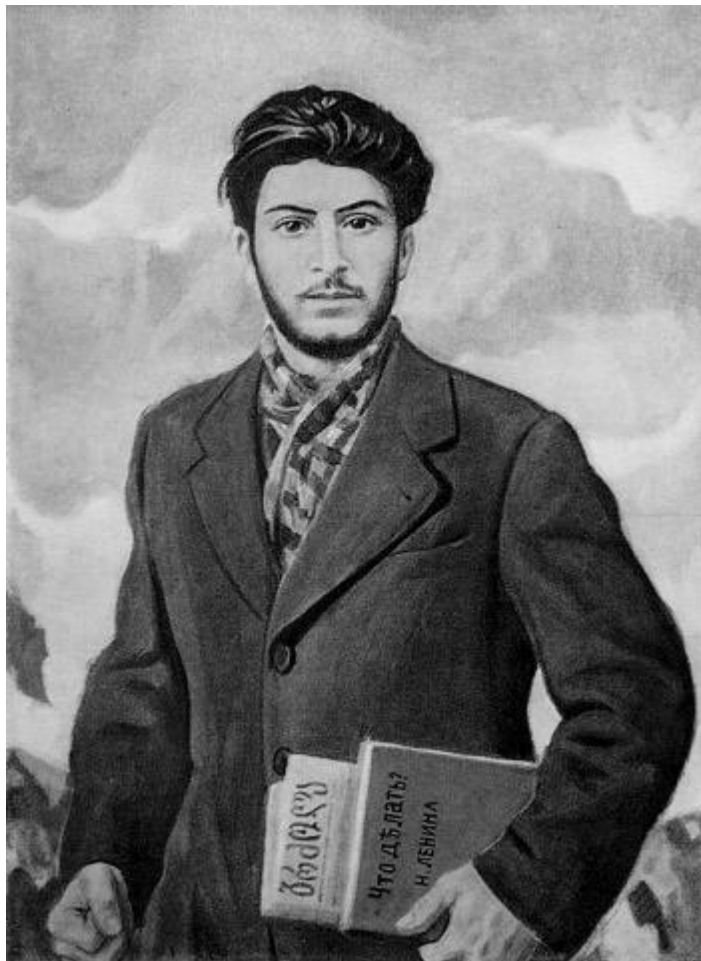
As funções a respeito de cada parte estão com o mesmo nome só que em inglês.

Parte 1)

1) Histograma

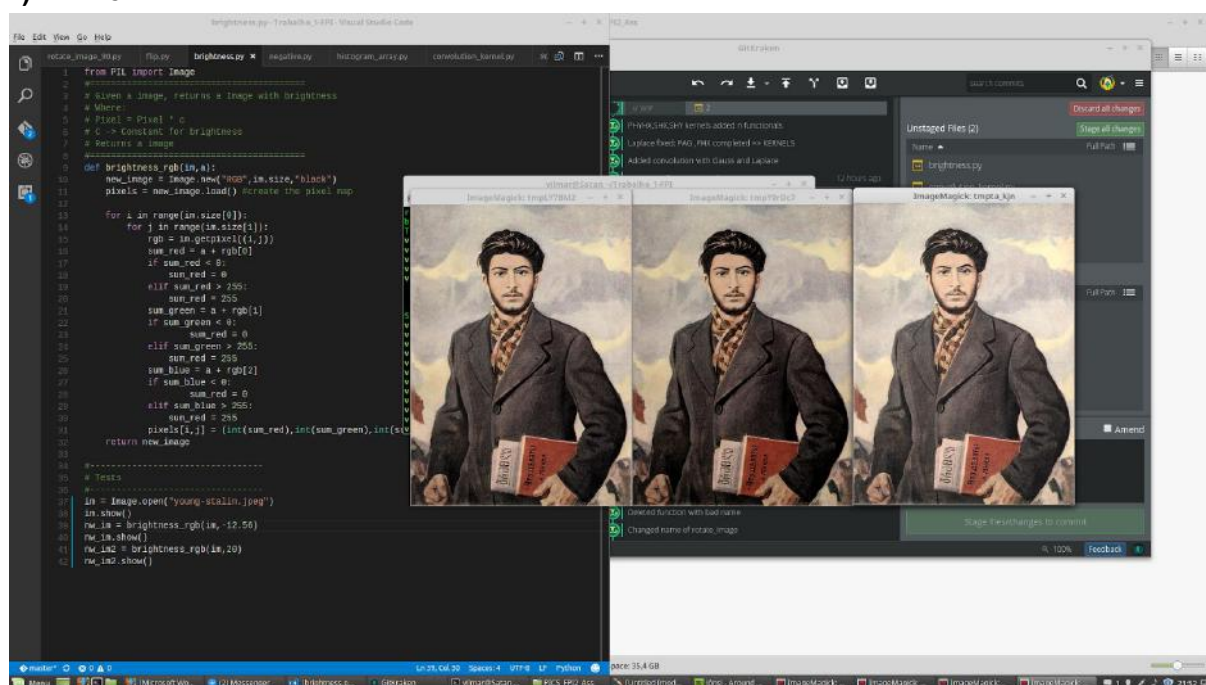


Este foi o histograma resultante da figura abaixo:



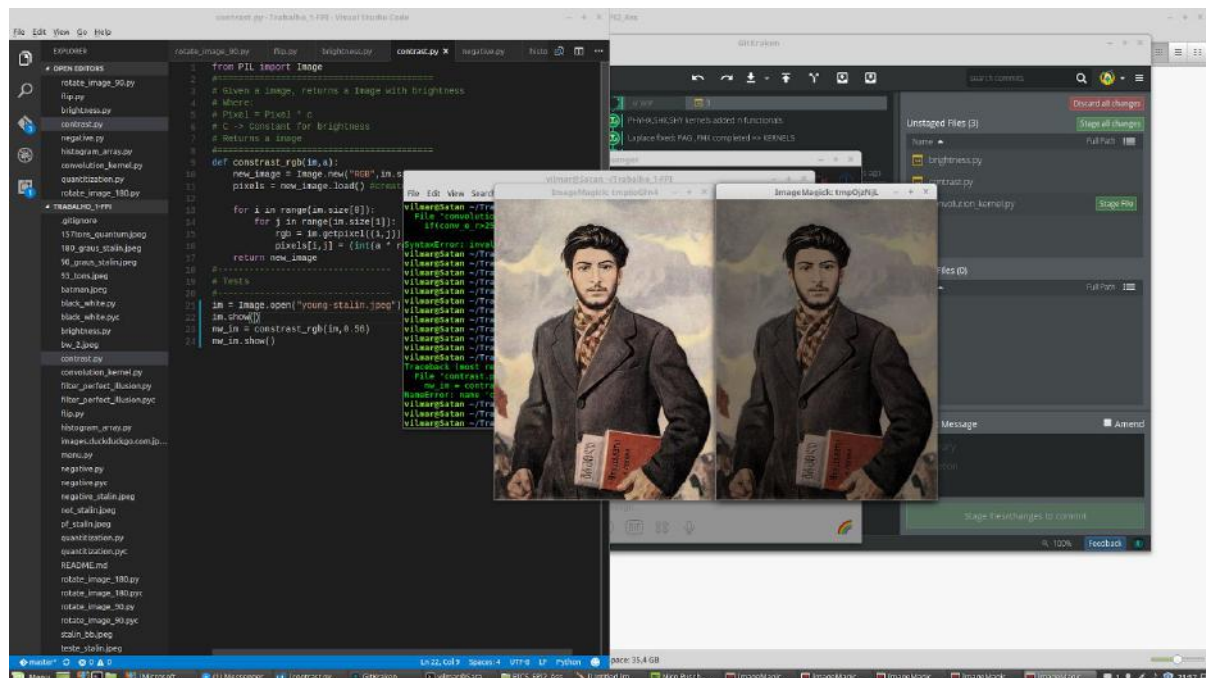
Foram feitas adaptações para exibir o histograma a partir do Matplotlib.

2) Brilho:



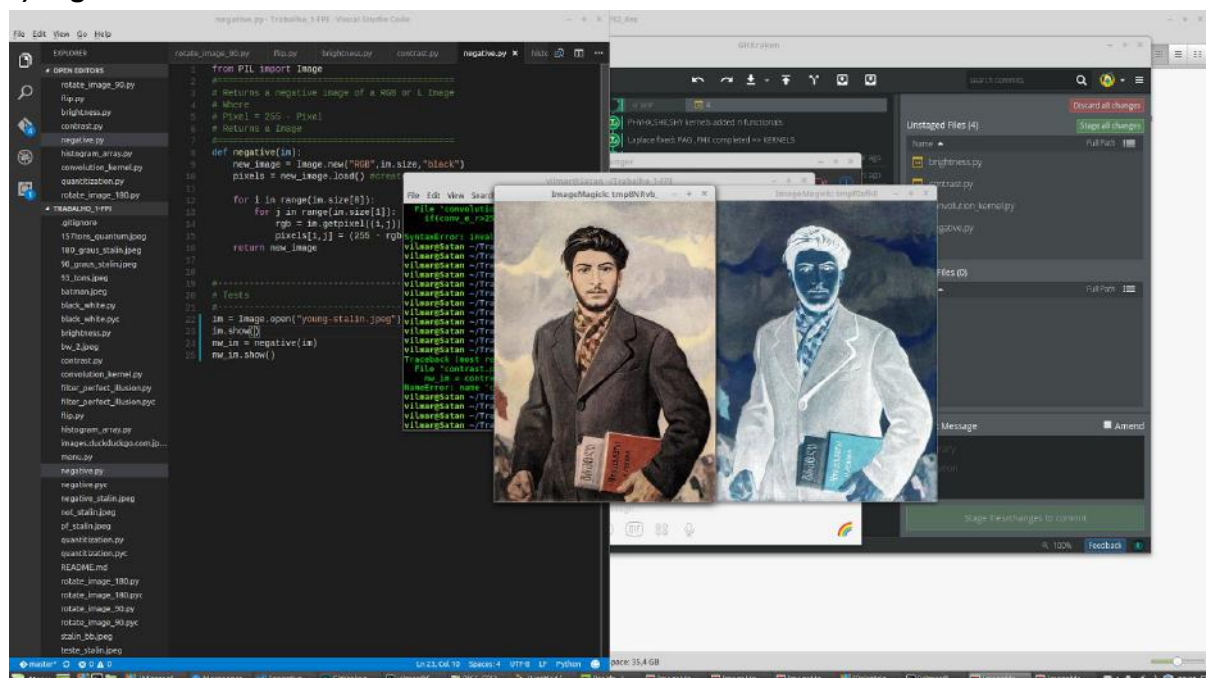
Na foto acima foram feitos testes, da esquerda para a direita, a imagem original, com brilho -12.56, com brilho em 20.

3) Contraste:



A imagem a seguir normal e com brilho em um valor entre (0,1), o valor foi de 0.56 e se nota como a imagem fica degradada.

4)Negativo:



Diminuindo 255 de cada pixel, se obtém o negativo. Acima estão a imagem original e seu negativo.

Parte 2)

8) Rotação de 90°

P.S.: A rotação de 90° já foi implementada anteriormente.

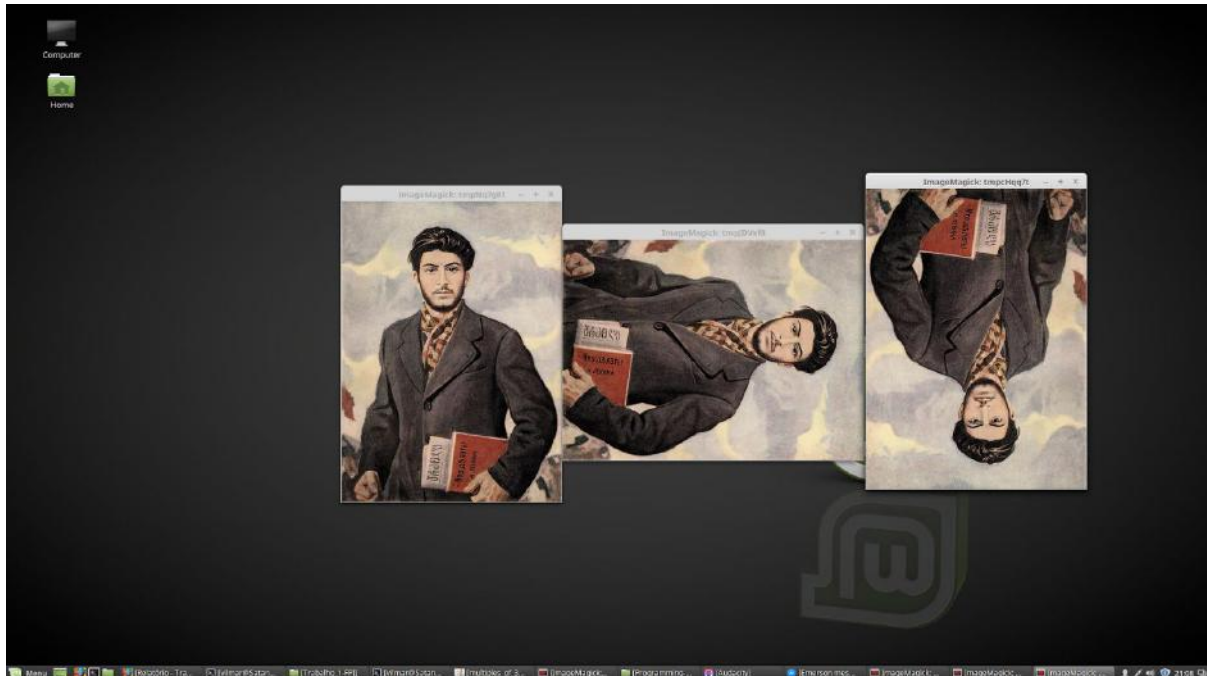
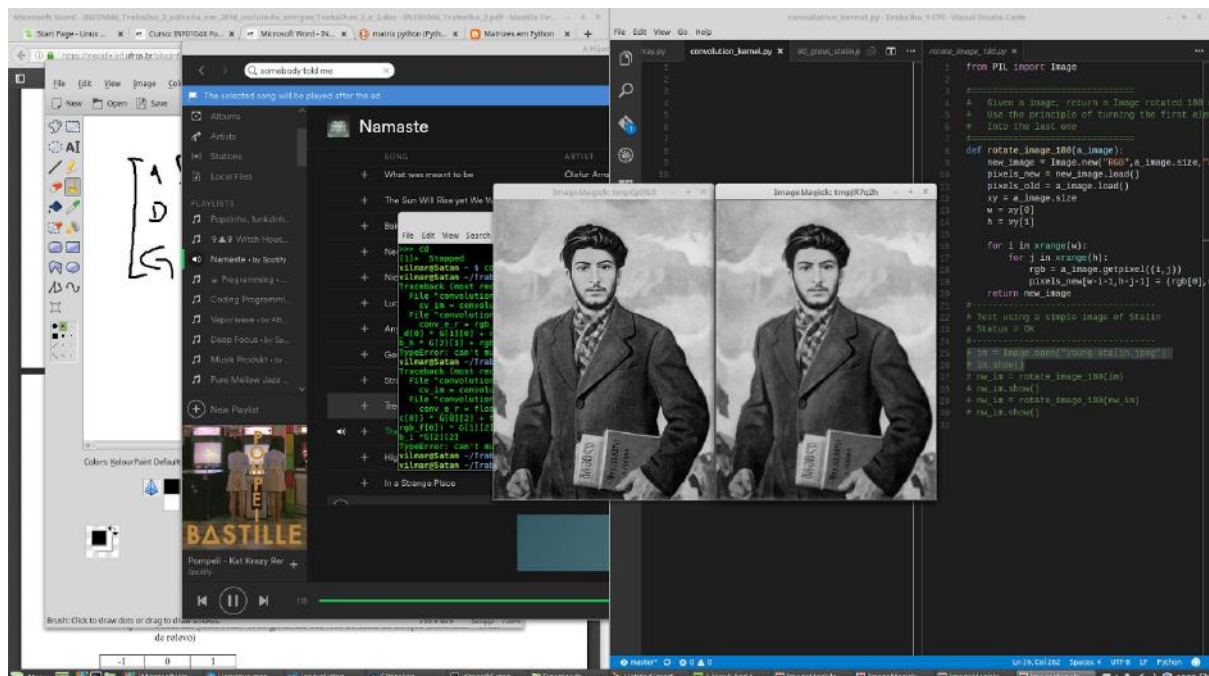


Imagem original, rotacionada 90° e rotacionada mais uma vez, ao todo 180° graus. Note que é possível rotacionar em 90° novamente a de 180°, permitindo ter 270°

9) Convolução e Kernels

Gaussiano:



Nota-se claramente o borrão na imagem, parecendo que esta se encontra distorcida, menos nítida.

Laplaciano:

The screenshot shows a computer screen with three main windows:

- Slide Window:** Displays a presentation slide titled "Laplaciano". It includes a diagram of a 3x3 grid with letters A through I, a 3x3 grid with numbers 1 through 9, and a 3x3 grid with letters A through I. Below the diagram, it lists three filters:
 - i) Gaussiana, filtro pass baixas (produz borramento).** A 3x3 grid with values: 0.6625, 0.125, 0.0625; 0.125, 0.25, 0.125; 0.0625, 0.125, 0.0625.
 - ii) Laplaciano, filtro pass altas (detecta arestas importantes).** A 3x3 grid with values: 0, -1, 0; -1, 4, -1; 0, -1, 0.
 - iii) Passa Altas Genérico (detector de arestas mais sensível).** A 3x3 grid with values: -1, -1, -1; -1, 8, -1; -1, -1, -1.
 - iv) Prewitt Hx (gradiente dos tons de cinza na direção horizontal - efeito de relevo).** A 3x3 grid with values: -1, 0, 1; -1, 0, 1; -1, 0, 1.
- Code Editor:** Shows Python code for image processing. The code includes a loop for `for i in range(1,w-1):` and `for j in range(1,h-1):`, followed by a series of `rgb` assignments and a `conv_e_r = int(conv_e_r)` statement. It also includes a `if conv_e_r > 255:` condition.
- Video Player:** Shows a video player with a progress bar and a play button. The video is titled "SEX AND LOVE" and is 4:02 long.

Passa Altas Genérico:

The screenshot shows a computer screen with three main windows:

- Slide Window:** Displays a presentation slide titled "Passa Altas Genérico". It includes a 3x3 grid with values: 0.6625, 0.125, 0.0625. Below the grid, it lists three filters:
 - ii) Laplaciano, filtro pass altas (detecta arestas importantes).** A 3x3 grid with values: 0, -1, 0; -1, 4, -1; 0, -1, 0.
 - iii) Passa Altas Genérico (detector de arestas mais sensível).** A 3x3 grid with values: -1, -1, -1; -1, 8, -1; -1, -1, -1.
 - iv) Prewitt Hx (gradiente dos tons de cinza na direção horizontal - efeito de relevo).** A 3x3 grid with values: -1, 0, 1; -1, 0, 1; -1, 0, 1.
- Code Editor:** Shows Python code for image processing. The code includes a loop for `for i in range(1,w-1):` and `for j in range(1,h-1):`, followed by a series of `rgb` assignments and a `conv_e_r = int(conv_e_r)` statement. It also includes a `if conv_e_r > 255:` condition.
- Video Player:** Shows a video player with a progress bar and a play button. The video is titled "SEX AND LOVE" and is 4:02 long.

Prewitt Hx:

The screenshot shows a computer screen with multiple windows. On the left, a web browser displays a document titled "Prewitt Hx" with the following content:

ii) **Passo Atlas Genérico** (detector de arestas mais sensível).

$$\begin{bmatrix} -1 & -1 & 0 \\ -1 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix}$$

b) **Prewitt Hx** (gradiente dos tons de cinza na direção horizontal - efeito de relevo)

$$\begin{bmatrix} -1 & -1 & -1 \\ -1 & 0 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

v) **Prewitt Hy** (gradiente dos tons de cinza na direção vertical - efeito de relevo)

$$\begin{bmatrix} -1 & 0 & 1 \\ 0 & 0 & 0 \\ 1 & 0 & -1 \end{bmatrix}$$

vi) **Sobel Hx** (mais sensível ao gradiente dos tons de cinza na direção horizontal - efeito de relevo)

$$\begin{bmatrix} -1 & 0 & 1 \\ 0 & 0 & 0 \\ 1 & 0 & -1 \end{bmatrix}$$

On the right, a code editor shows Python code for image processing. The code includes comments and function calls for image processing. The image processing windows show the original image and its edge detection result using the Prewitt Hx kernel.

Prewitt Hy Hx:

The screenshot shows a computer screen with multiple windows. On the left, a web browser displays a document titled "Prewitt Hy Hx" with the following content:

Necklaces, Convolutions, and X-Y - Jeff Erickson

Necklaces, Convolutions, and X-Y - researchgate.net

Necklaces, Convolutions, and X-Y - ResearchGate

MATLAB Central - 2D convolution with unit-sum kernel

Necklaces, Convolutions, and X-Y

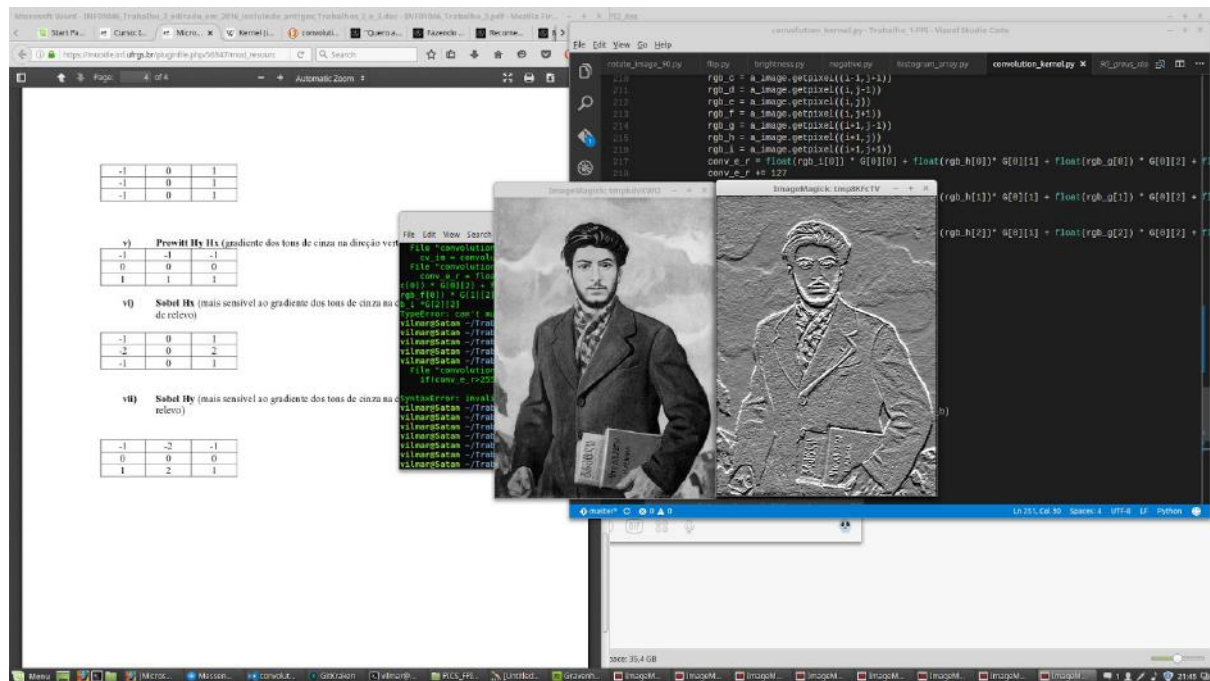
* Elvis Presley (Gospel) - YouTube

Lecture 5 - scribd.com

sec14.htm - University of Cambridge

On the right, a code editor shows Python code for image processing. The code includes comments and function calls for image processing. The image processing windows show the original image and its edge detection result using the Prewitt Hy Hx kernel.

Sobel Hx:



Sobel Hy:

