

Universidade Federal do Rio Grande do Sul

Instituto de Informática

INF01046 Fundamentos de Processamento de Imagens

Professor Manuel M. Oliveira

Relatório - Laboratório 01

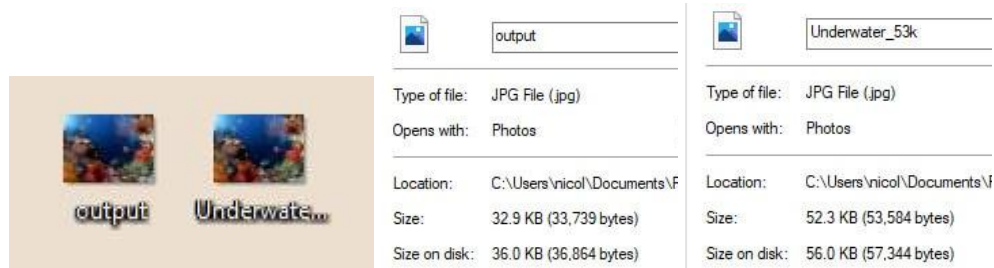
2020/2

Nicolle Pimentel Favero

Parte I – Leitura e Gravação de Arquivos de Imagens

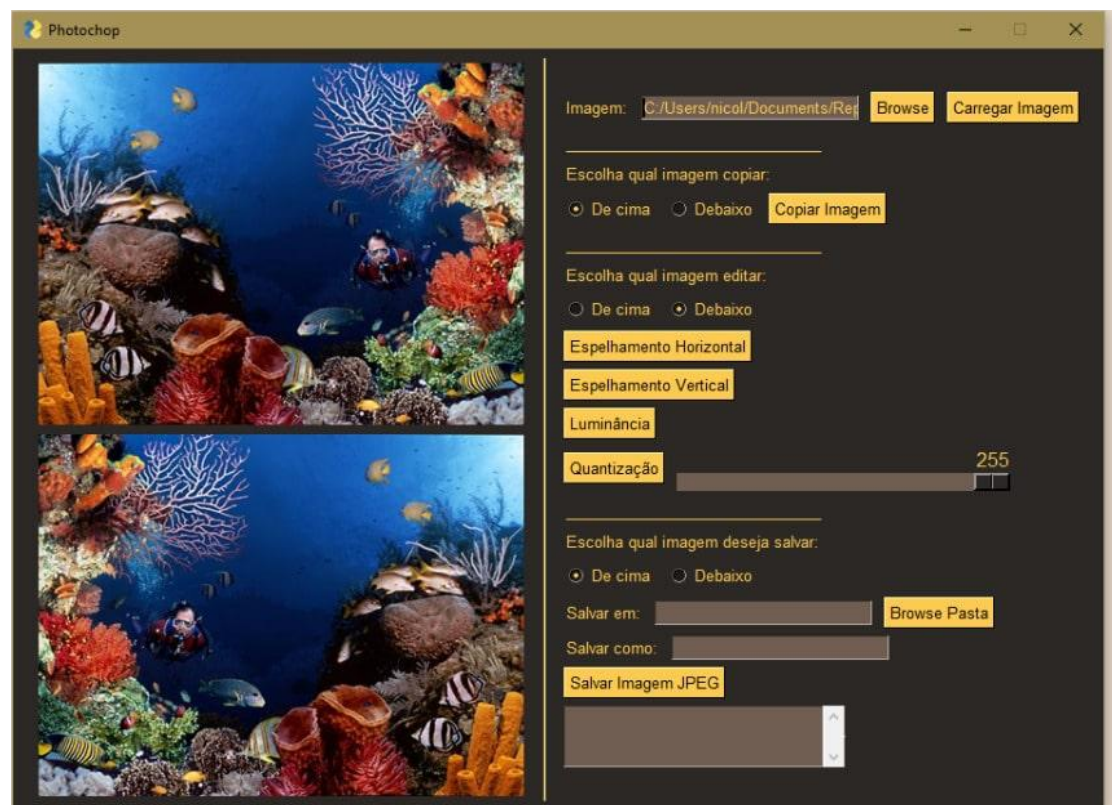
Pontos Observados:

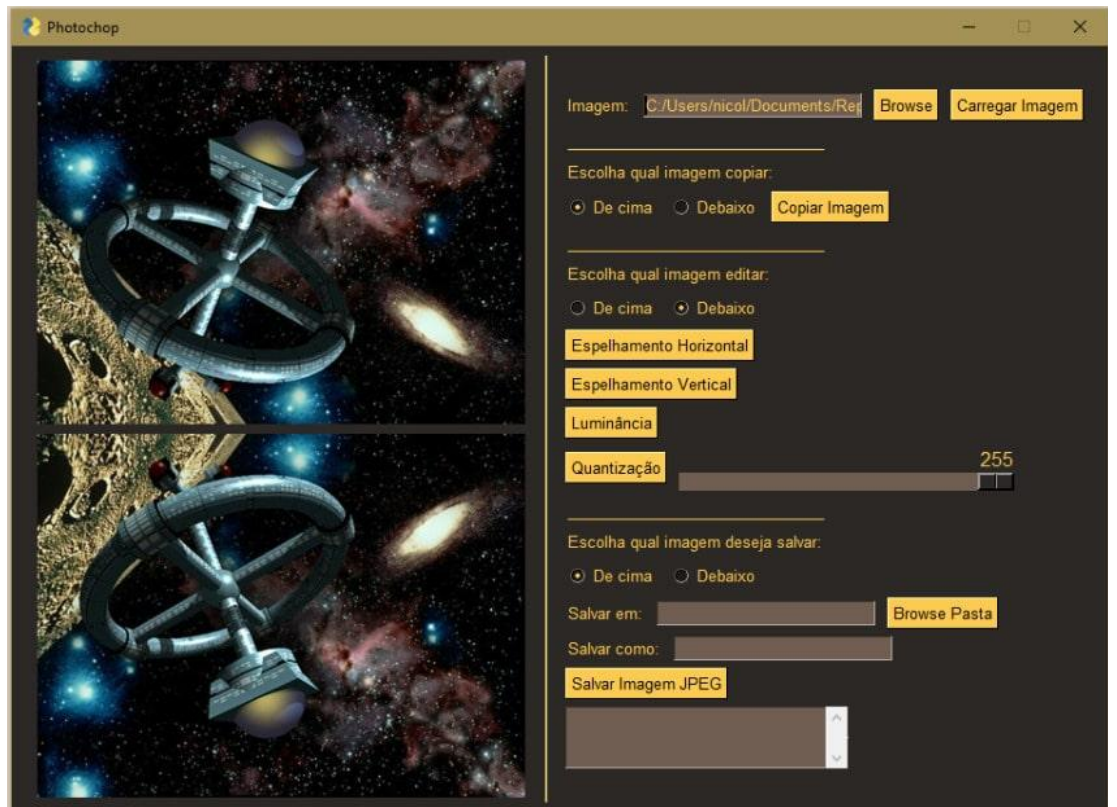
1. Os arquivos gravados pelo programa têm menor qualidade e menor tamanho em comparação às imagens originais fornecidas para o trabalho. Isso por que a maneira como o JPEG é salvo inclui compactar o tamanho do arquivo descartando dados não tão importantes.



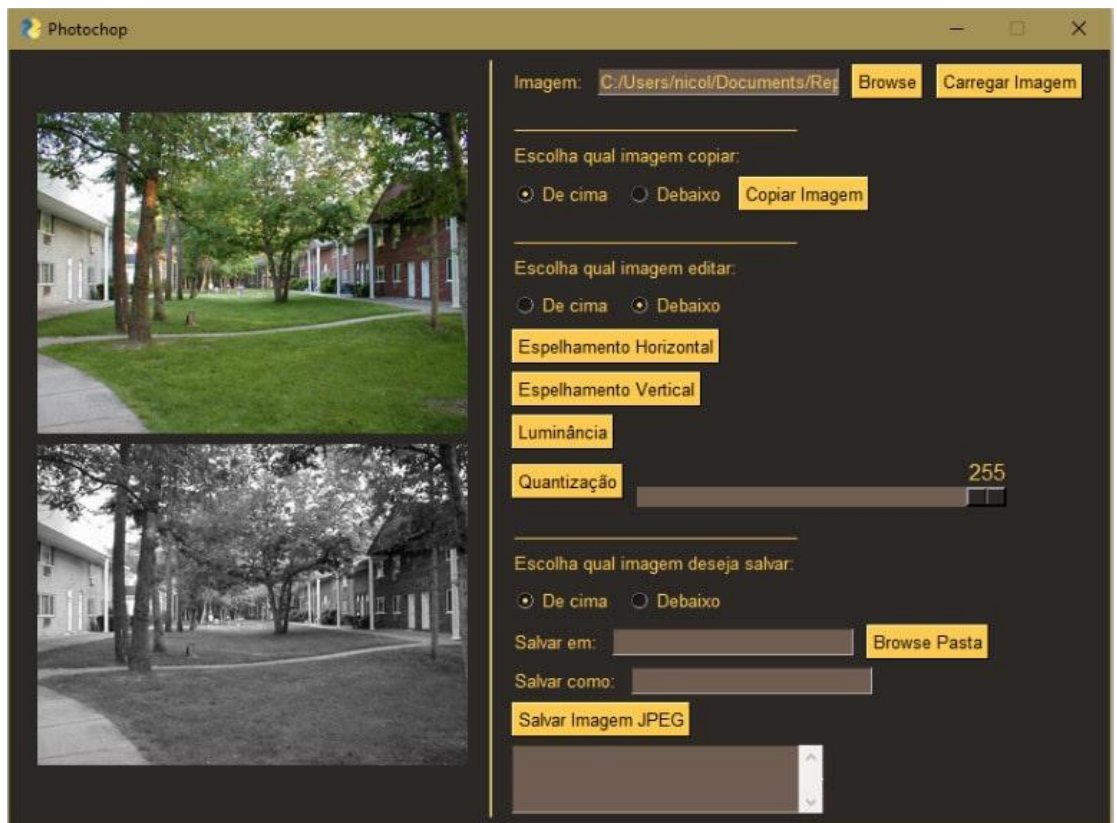
Parte II – Leitura, Exibição e Operações sobre Imagens (80 pontos)

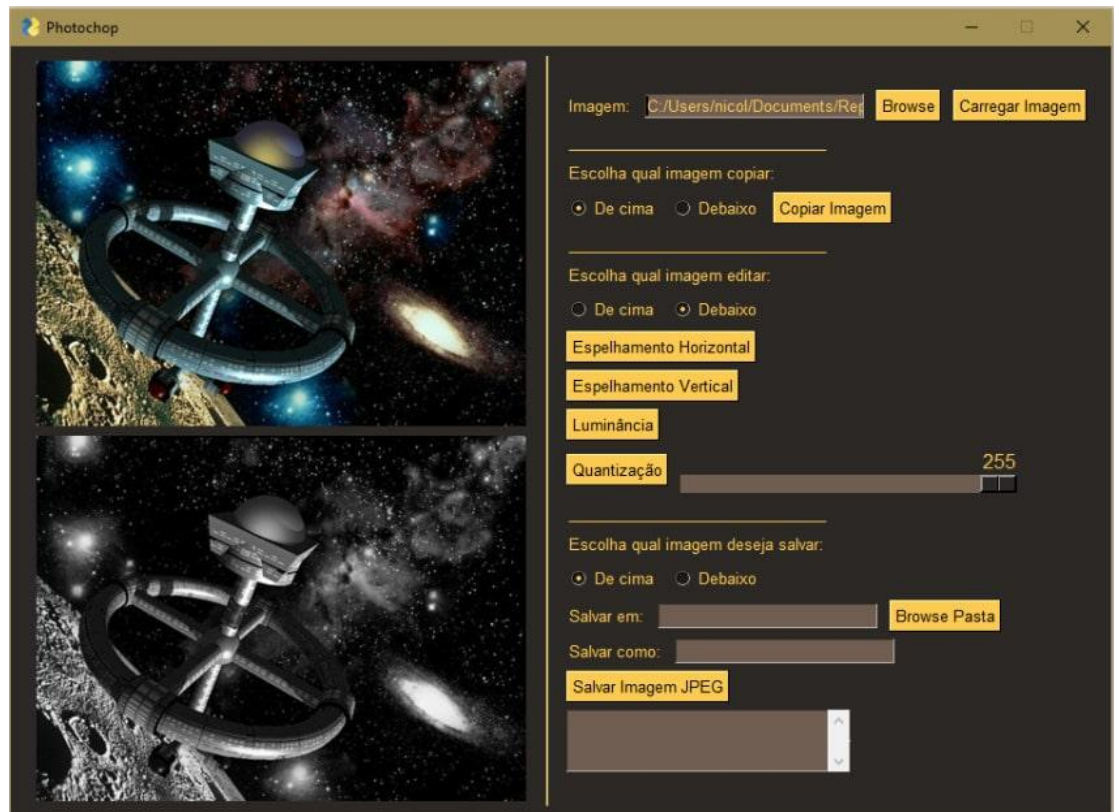
1. Operações
 - a. Espelhamento Horizontal e Vertical:





b. Luminância:





Caso o cálculo de luminância seja aplicado várias vezes a uma mesma imagem que já esteja em tons de cinza, não percebe-se nenhuma alteração.

c. Quantização:

Para a operação de quantização foi utilizada uma equação mais simples do que a deduzida pelo enunciado.

Equação deduzida do enunciado:

$$\text{indice} = (2 * \text{pixel} - 2 * \text{tom_minimo} + 1 - \text{tamanho_bin}) / (2 * \text{tamanho_bin}),$$

onde o índice em questão seria a tendência do índice que indica a qual intervalo o pixel de origem pertence. O problema encontrado foi que a função ora era necessário arredondar o índice para cima, ora era necessário arredondar o índice para baixo e, com isso, muitas vezes era obtido o índice errado do intervalo no qual o pixel deveria ser mapeado.

Equação mais conveniente:

$$\text{indice} = \text{floor}((\text{pixel} - \text{tom_minimo}) / \text{tamanho_bin})$$

pois a equação em si sempre resulta em um índice um pouco maior e, com o arredondamento para baixo, resulta sempre (pelo menos em todos os testes e tentativas feitas) em um índice correto.

n = 70



n = 30



n = 5



d. Salvar Imagem em JPEG:



2. Sobre o trabalho

Todas as etapas foram completadas conforme solicitado pelo professor no enunciado do trabalho. Uma dos poucos requisitos que não foi cumprido foi o de colocar as imagens lado a lado na interface. Optei por exibir as imagens em coluna em função de uma melhor adaptação da interface à tela e a fim de proporcionar um espaço adequado ao menu. Acredito que isso não seja um grande problema, tendo em vista a funcionalidade do programa como um todo e a forma como ele faz a comunicação com o usuário.

Uma dificuldade inicial foi compreender o objetivo do trabalho, de ter que implementar essas operações manualmente e não de usar métodos prontos de alguma biblioteca ou API.

Também encontrei bastante dificuldade na função da quantização, embora o enunciado desse várias informações sobre os intervalos e sobre os condicionais da operação. Buscando mais informação na internet e entendendo sobre histogramas, consegui entender melhor a quantização e seu funcionamento.

Outro desafio foi mexer com GUI pela primeira vez. Nunca foi uma prioridade até então, mas foi uma experiência muito divertida.

Quanto às coisas que faria diferente no decorrer do trabalho seria não ter deixado para fazer no final do prazo. Ocorreu tudo bem, apesar das dificuldades, mas creio que esses obstáculos não teriam sido tão cansativos se tivesse aproveitado melhor o espaçamento do prazo do trabalho.

3. Sobre a interface

Para a implementação do trabalho, foi escolhido Python 3.9, uma linguagem com a qual já tenho certa familiaridade, mas na qual não tenho “fluência”. Por causa do fato de nunca ter mexido com GUIs, optei por uma GUI simples e fácil de mexer, a PySimpleGUI. Vendo o resultado final, acredito que a interface tenha ficado intuitiva, conforme solicitado no enunciado, embora aquela inversão na disposição das imagens mencionada antes. Priorizei o entendimento do usuário sobre qual das cópias da imagem ele estaria manipulando através de estruturas Radio com configurações default, além da divisão das etapas do processamento de imagem no projeto. As imagens são mostradas uma em cima da outra à esquerda do menu.

