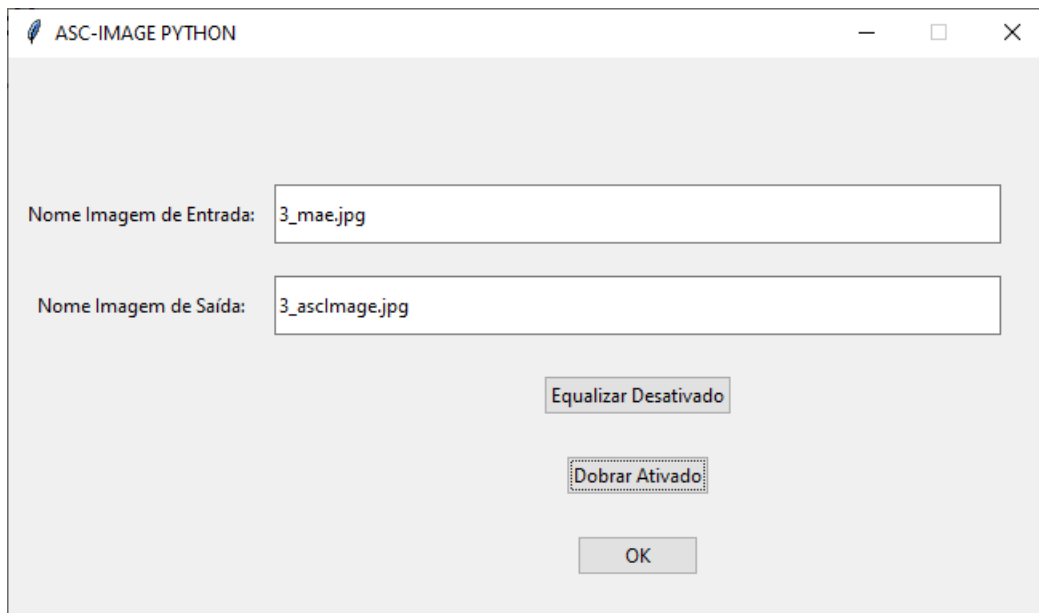


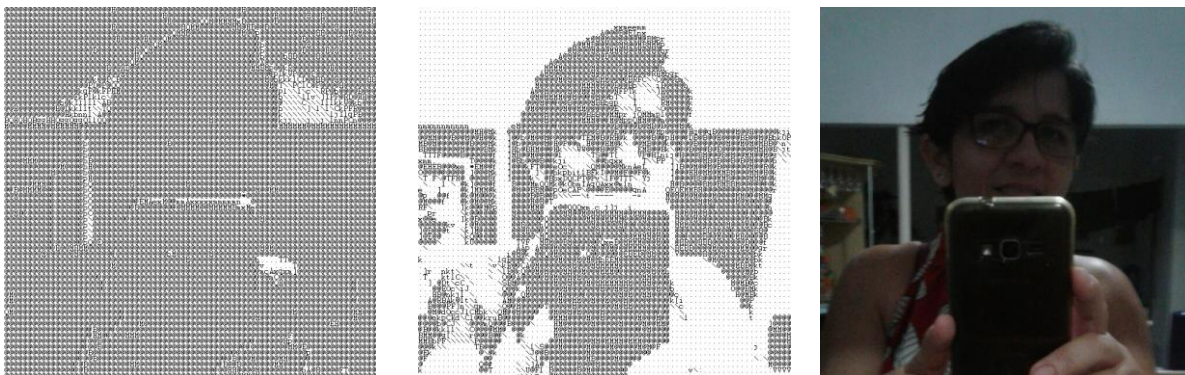
Trabalho: ASC-Image

Alunos: Flávio Correia e Pedro Leonardo



Os caminhos dos campos de entrada devem ser dados a partir de onde o arquivo python está. Em ambos os campos o formato da imagem deve ser especificado.

O primeiro botão possui duas opções, equalizar ativado **Equalizar Ativado**, que faz a equalização da imagem, ou equalizar desativado **Equalizar Desativado**. A equalização chega a melhorar muito o resultado final da imagem.



A primeira imagem é feita com o processamento sem equalização e dobrar desativado. Já a segunda é feita com equalização e dobrar desativado. A imagem original é a última da direita.

Quanto ao segundo botão ele pode estar em três estágios.

**Dobrar Desativado**, não dobra a imagem original.

**Dobrar Ativado**, dobra a imagem de entrada para que o número de caracteres utilizados para representar uma parte da imagem aumente também, melhora o aspecto geral da imagem, como ponto negativo a imagem de saída tem o dobro da imagem de entrada.



**Dobrar Ativado com Redução**

, faz a mesma coisa que o botão de Dobrar Ativado mas ao final retorna a imagem para o seu tamanho original, como ponto negativo a qualidade das letras é prejudicado.

Imagem sem equalização e dobrar desativado.



Imagem com equalização e dobrar desativado.





Imagem com equalização e dobrar ativado.



Imagem com equalização e dobrar com redução ativado.



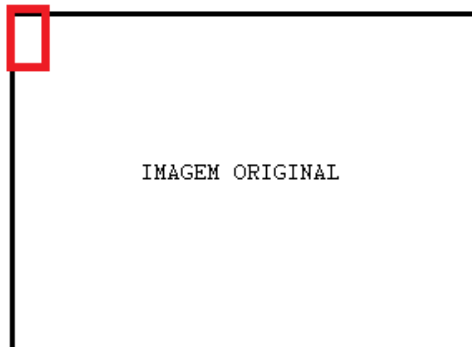
Imagem original.



Sobre o algoritmo:

Nosso algoritmo segue uma implementação “burra” da criação de imagens ASCII, ele seleciona uma parte da imagem correspondente a 8x10 pixels e compara essa área com as imagens dos caracteres contidos na pasta asclimages, que contém 72 imagens, ele faz o somatório do erro absoluto e escolhe o caractere que corresponde a imagem de menor erro e coloca essa imagem na imagem final ASCII, ele percorre toda a imagem fazendo esse procedimento. Como segue a explicação da imagem.

1º  
Área de análise de tamanho 8x10



2º

Compara a imagem de análise com todas as imagens de caracteres, somando o valor absoluto da diferença dos pixel da imagem de análise e a de comparação.

A	erro = 528
B	erro = 327
⋮	
@	erro = 472

3º

A imagem de comparação com o menor erro é a que ocupará posição da imagem de análise. O B no exemplo.

4º

Fazer esse procedimento para toda a imagem.

A função `criaAI(img)` é a função que gera a imagem ASC. Ela recebe a imagem de entrada como `img`, cria uma matriz `imgF` de uns que é do tipo `uint8`, depois percorre `img` atribuindo as imagens dos caracteres que tem o menor erro e representam aquela área na `imgF`. Ao final ela é retornada.

A função de equalização e redimensionamento é feita na função `preparaimagem()` que está dentro da função `janela()`, que é responsável pela interface gráfica, feita utilizando o `tkinter`.