

Trabalho 1 de Introdução ao Processamento de Imagens

Victor André Gris Costa
Universidade de Brasília - UnB
Departamento de Ciências da Computação
Brasília, Brasil
victorandr98@gmail.com

Resumo—Este trabalho tem como o objetivo a fixação de conceitos básicos de processamento de imagens através da resolução de 2 problemas: mudar o tamanho e profundidade de uma imagem e achar o caminho de menor esforço entre dois pontos de uma imagem da superfície de Marte. No primeiro problema foi resolvida a mudança de profundidade descartando os bits menos significativos. Para a mudança de escala foi feita a interpolação pelo pixel mais próximo quando as dimensões aumentavam e média dos pixels mais próximos quando decresciam. No segundo problema a imagem foi transformada em escala de cinza, fez-se a equalização do histograma e por fim avaliou-se o menor caminho a partir dos níveis de brilho e distância euclidiana dos pixels da imagem equalizada. Os resultados foram positivos para o primeiro problema, conseguindo mudar a escala para qualquer proporção e também diminuir a quantidade de níveis da imagem. Para o segundo problema obteve-se resultados bons, conseguindo obter o caminho de menor esforço.

I. INTRODUÇÃO

Processamento de imagens é uma subárea de processamento digital de sinais na qual são aplicados processos em imagens para obter novas imagens. Hoje em dia com o maior poder de processamento esta área se tornou essencial para muitas áreas. Por exemplo na medicina ele é utilizado para melhorar imagens médicas. Neste trabalho serão utilizados alguns conceitos básicos e importantes para o processamento de imagens. Eles são a profundidade de uma imagem, escala de cinza, histograma e equalização.

A profundidade de uma imagem é caracterizada pela quantidade de brilhos possíveis para um canal de cor só. Geralmente essa profundidade é de 256 níveis. Ela também pode ser medida pelo número de *bits* que contam qual o nível. Neste caso a profundidade será igual a 2^N , sendo N o número de bits do canal.

Uma imagem pode ser capturada com níveis de brilho para 3 cores primárias, vermelho, verde e azul, ou só os brilhos absolutos. No segundo caso diz-se que a imagem está em escala de cinza. Nós podemos a partir de uma imagem colorida obter a imagem em escala de cinza utilizando uma média ponderada dos canais. Como os humanos conseguem ver melhor o brilho da luz verde o peso da cor verde será maior que os pesos da cor vermelha e cor azul.

O histograma é uma ferramenta vinda da estatística que é basicamente a contagem de quantas vezes um certo elemento aparece. No caso do processamento de imagens esse processo é aplicado contando quantas vezes cada nível de brilho aparece

em sua imagem. Este processo é utilizado para descobrir propriedades da imagem, como por exemplo se a imagem é escura ou clara ou se tem alto ou baixo contraste, e também para melhorar imagens como a equalização de histograma, que permite utilizar todo o contraste possível e perceber mais detalhes.

A equalização de histograma é uma técnica para utilizar os níveis de brilho da melhor forma possível. Ele funciona deixando o histograma o mais próximo de uniforme o possível. É um método muito utilizado em imagens para evidenciar detalhes escondidos.

II. METODOLOGIA

A. Questão 1

Foi proposto que fosse criada uma função para mudar as dimensões e profundidade de uma imagem. A função deve ser compatível com imagens em escala de cinza e imagens coloridas. A função deve receber 2 parâmetros, um dizendo quantos bits por canal deverá ter a nova imagem, alterando a profundidade, e outro um valor decimal indicando quantas vezes maior será a nova imagem. Uma imagem com 5 bits deverá ter 32 níveis de cores e uma imagem escalonada pelo fator 2.0 deverá ter o dobro de largura e altura. caso o fator de escalonamento seja menor que 1, a imagem deverá diminuir, por exemplo para um fator de 0.5 a imagem terá metade da largura e altura.

Para a resolução deste problema foi criada apenas uma função nomeada *im_chscaledepth*. Nesta função foram implementadas 3 blocos principais de código, um para o aumento das dimensões da imagem, um para a redução das dimensões da imagem e um para reduzir a profundidade da imagem.

No bloco de aumento da imagem começa-se criando uma imagem preta com as novas dimensões. Logo após percorre-se a imagem original copiando os valores de cada pixel da imagem original em um bloco de $scale \times scale$ pixels, sendo $scale$ o fator multiplicativo das dimensões passado, da nova imagem. Para escalas de aumento não inteiras não criarem uma grade na imagem, os arredondamentos de números de ponto flutuante para inteiros só ocorre após todas as contas de limites inferiores e superiores de cada bloco da imagem. Quando o fator de escala era arredondado antes de determinar os limites, grades pretas eram geradas na imagem.

Para a redução da imagem o algoritmo é de certa forma inverso ao de aumento. Primeiro cria-se uma nova imagem com as novas dimensões. Depois calcula-se o inverso da escala, que chamaremos de *fac*. Este fator representa quantas vezes a imagem original é maior que a nova. Por fim percorre-se a matriz nova pixel a pixel e a eles são atribuídos a média dos valores do bloco de $fac \times fac$ respectivo àquele pixel.

Após esses procedimentos realiza-se a redução da profundidade da imagem. Primeiramente são retirados os $8 - n_bits$ bits menos significativos do canal da imagem, sobrando apenas os n_bits mais significativos. A escolha dos bits menos significativos se deve a eles serem sempre o bits de menor peso no nível de brilho. Após esse processo a nova imagem possui a quantidade de níveis desejados, porém por não utilizar todos os 8 bits possíveis, a exibição da imagem fica muito escura. Para resolver esse problema encontra-se um fator multiplicativo que faz com que o nível mínimo seja 0, o nível máximo seja 255 e faça com que todos os níveis sejam igualmente espaçados uns dos outros. Fazendo as contas encontra-se que o fator multiplicativo deve ser $255/2^{n_bits} - 1$, que é basicamente o valor máximo para 8 bits dividido pelo valor máximo para n_bits .

Para o tratamento de imagens poderem estar em escala de cinza ou coloridas, basicamente faz-se com que as imagens tenham a mesma quantidade de dimensões. Caso a imagem seja colorida, ela é mantida intacta por ser uma matriz de três dimensões. Caso esteja em escala de cinza, a imagem será uma matriz de duas dimensões. Para resolver isso converte-se a matriz bidimensional em uma tridimensional cuja terceira dimensão só tenha uma unidade. Passam-se as imagens pelo algoritmo e ao fim as matrizes cujas terceiras dimensões tenham apenas uma camada são convertidas em matrizes bidimensionais.

B. Questão 2

Foi proposto que a partir de uma imagem de parte da superfície de Marte e sabendo onde a equipe de astronautas da primeira missão a Marte está, deve-se encontrar o caminho de menor esforço até o ponto de coleta de material de estudo. Para isso foram fornecidas uma imagem *bitmap* de parte da superfície de Marte, um ponto de começo e um ponto de fim. Foi também informado que o esforço será medido por distância euclidiana do fim e brilho do pixel.

A resolução desenvolvida consiste na implementação do seguinte algoritmo:

- 1) Converta a imagem da superfície de Marte de colorida para escala de cinza;
- 2) Realize a equalização de histograma da imagem da superfície de Marte em escala de cinza;
- 3) A partir da imagem equalizada, pixel de início e pixel de fim, calcule o caminho de menor esforço;
- 4) Desenhe o caminho de menor esforço na imagem original.

Para cada uma das etapas apresentadas criou-se uma função implementando-a. No procedimento principal foram feitas as chamadas às funções e as passagens de parâmetros, permitindo o reuso para futuros projetos.

Para a primeira etapa criou-se a função *convert2grayscale* que recebe uma imagem colorida como parâmetro. Para a conversão apenas separaram-se cada um dos canais de cor e faz-se uma média ponderada dos canais pixel a pixel. No caso o verde possui um peso maior por conta dos humanos perceberem a cor verde melhor.

Na segunda etapa é feita a equalização de histograma. Foi criada uma função chamada *equalization* que recebe a imagem em escala de cinza. Para realizar a equalização primeiro calcula-se o histograma da imagem e depois normaliza-se ele. A partir do histograma normalizado calcula-se a função de distribuição acumulada. Os valores acumulados são arredondados para o inteiro mais próximo. A partir dessa distribuição acumulada consegue-se obter uma função que conecta o nível original de brilho com o novo nível de brilho. O novo nível será o valor da distribuição acumulada arredondada para o inteiro mais próximo. Essa função é aplicada na imagem original e cria-se uma nova imagem que tem o histograma o mais próximo de equalizado o possível.

Para encontrar o caminho de menor esforço criou-se a função *go2destination* que recebe a imagem equalizada, o ponto inicial e o ponto final e retorna um vetor contendo o caminho. Primeiro começa-se processando o pixel inicial, os 8 vizinhos desse pixel têm a distância euclidiana até o pixel de destino calculados e são selecionados os com as 3 menores distâncias. Os 3 pixels selecionados têm seus brilhos analisados e seleciona-se o de menor brilho. O pixel escolhido é guardado como parte do caminho e repete-se o processo até chegar no pixel de destino.

III. RESULTADOS

A. Questão 1

Realizando os procedimentos explicados acima nas figuras 1 e 4 obtivemos as imagens a seguir. Vale notar que as imagens apresentadas não estão na mesma escala para poder ter uma melhor visualização delas.



Figura 1. Imagem original colorida.



Figura 2. Imagem obtida após passar a figura 1 pela função `im_chscaledepth` com os parâmetros de 5 bits de profundidade e 0,50 de escala.



Figura 5. Imagem obtida após passar a figura 4 pela função `im_chscaledepth` com os parâmetros de 5 bits de profundidade e 0,50 de escala.



Figura 3. Imagem obtida após passar a figura 1 pela função `im_chscaledepth` com os parâmetros de 3 bits de profundidade e 1,75 de escala.



Figura 6. Imagem obtida após passar a figura 4 pela função `im_chscaledepth` com os parâmetros de 3 bits de profundidade e 1,75 de escala.

B. Questão 2

Aplicando o procedimento descrito acima obtivemos as imagens da superfície de Marte a seguir. Vale ressaltar que houve compressão JPEG nas imagens, pois as originais em formato *bitmap* eram grandes demais para este relatório.



Figura 4. Imagem original convertida para escala de cinza.

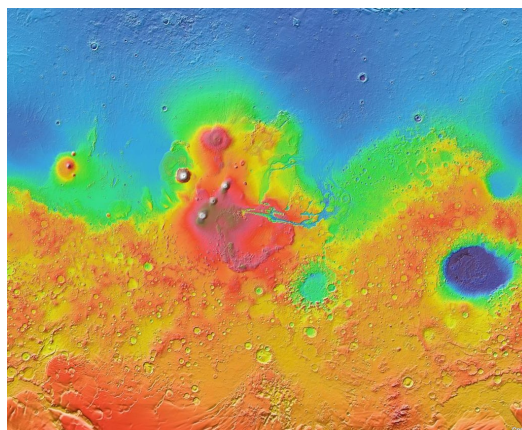


Figura 7. Imagem original da superfície de Marte.

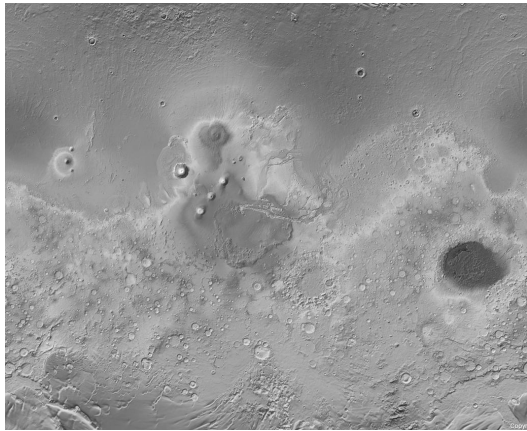


Figura 8. Imagem convertida em escala de cinza da superfície de Marte.

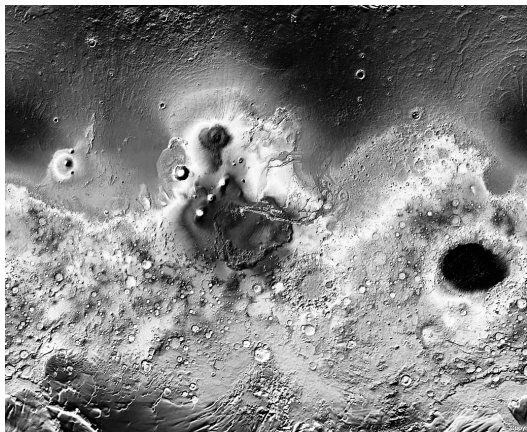


Figura 9. Imagem com o histograma equalizado da superfície de Marte.

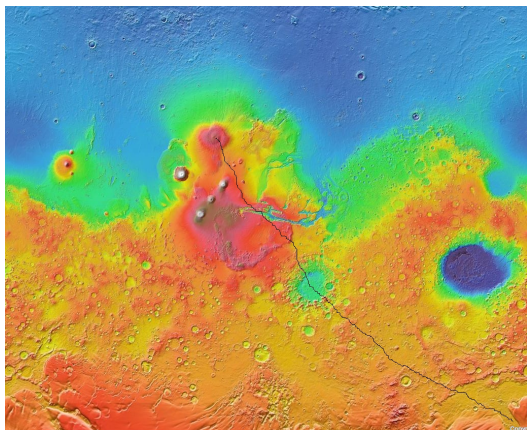


Figura 10. Imagem da superfície de Marte com o caminho de menor esforço.

IV. CONCLUSÃO

Ao fim das questões é possível constatar algumas informações interessantes.

Nas figuras 2 e 5 é possível ver que não houve muitas perdas por diminuir a profundidade e que é um mecanismo que pode ser utilizado para comprimir imagens. No entanto não pode exagerar, pois é muito notável a perda qualitativa nas imagens 3 e 6 por conta da redução da quantidade de níveis de brilho.

Também foi possível aprender que ao mudar a escala de uma imagem é importante tomar cuidado com quando são realizados os arredondamentos dos números reais.

Inferiu-se a partir da questão 2 que a equalização de histograma é muito útil para realce de detalhes escondidos nada visíveis nas imagens colorida e em escala de cinza.

REFERÊNCIAS

- [1] Slides de Aula do professor Bruno Macchiavello.
- [2] "OpenCV: OpenCV modules", Docs.opencv.org, 2019. [Online]. Disponível: <https://docs.opencv.org/4.1.0/>. [Acessado: 21- Abril- 2019]
- [3] "NumPy Reference — NumPy v1.16 Manual", Docs.scipy.org, 2019. [Online]. Disponível: <https://docs.scipy.org/doc/numpy/reference/>. [Acessado: 21- Abril- 2019]