

Trabalho 2

1 Especificação do Problema

O propósito deste trabalho é realizar experimentos com métodos de limiarização global e local em imagens monocromáticas. Métodos de limiarização global determinam um único valor de limiar para toda a imagem. Por outro lado, métodos de limiarização adaptativa local calculam um limiar para cada pixel da imagem com base na informação contida em uma vizinhança do pixel. Em ambos os casos, se um pixel $p(x, y)$ na imagem de entrada possuir um valor mais alto que o limiar, então o pixel $p(x, y)$ é classificado como objeto (preto), caso contrário, ele é rotulado como fundo (branco).

Implemente métodos de limiarização global e local, conforme descrição abaixo, aplicando-os à imagem de entrada.

- Método Global: se o valor de intensidade de um pixel (x, y) for maior do que um limiar T (por exemplo, $T = 128$), o pixel será considerado como pertencente ao objeto; caso contrário, será considerado como fundo.
- Método de Otsu [3]: calcula automaticamente um limiar ótimo para dividir a imagem em duas classes de pixels por meio da minimização da variância intraclasse.
- Método de Bernsen [1]: para cada pixel (x, y) , o limiar é calculado como

$$T(x, y) = (z_{\min} + z_{\max})/2 \quad (1)$$

em que z_{\min} e z_{\max} são os valores de níveis de cinza mínimo e máximo, respectivamente, em uma vizinhança $n \times n$ centrada em (x, y) .

- Método de Niblack [2]: o valor de limiar é baseado na média local e no desvio padrão. O valor de limiar em um pixel (x, y) é calculado como

$$T(x, y) = \mu(x, y) + k \sigma(x, y) \quad (2)$$

em que $\mu(x, y)$ e $\sigma(x, y)$ são a média e o desvio padrão, respectivamente, em uma vizinhança local de (x, y) . O tamanho da vizinhança deveria ser suficientemente pequeno para preservar detalhes locais, mas, ao mesmo tempo, suficientemente grande para suprimir ruído. O valor de k é usado para ajustar a fração da borda do objeto a ser considerada como parte do objeto.

- Método de Sauvola e Pietaksinen [5] procura melhorar os resultados do método de Niblack, particularmente para imagens de documentos apresentando má iluminação. O limiar se adapta de acordo com a média local e o desvio padrão sobre uma janela de tamanho $n \times n$ pixels. O limiar em um pixel (x, y) é calculado como

$$T(x, y) = \mu(x, y) \left[1 + k \left(\frac{\sigma(x, y)}{R} - 1 \right) \right] \quad (3)$$

em que $\mu(x, y)$ e $\sigma(x, y)$ são definidos como em [2]. Sauvola e Pietaksinen sugerem valores de $k = 0.5$ e $R = 128$.

- Método de Phansalskar, More e Sabale [4]: variação do método [5] para lidar com imagens de baixo contraste. O limiar é calculado como

$$T = \mu(x, y) \left[1 + p \exp(-q \mu(x, y)) + k \left(\frac{\sigma(x, y)}{R} - 1 \right) \right] \quad (4)$$

em que $\mu(x, y)$ e $\sigma(x, y)$ são a média e o desvio padrão, respectivamente, em uma vizinhança local de (x, y) . Os autores sugerem $k = 0.25$, $R = 0.5$, $p = 2$ e $q = 10$. Qualquer outro valor numérico diferente de 0 é válido para k . O valor de R é diferente do método [5], pois usa uma intensidade normalizada da imagem.

- Método do Contraste: atribui o valor de um pixel como fundo ou objeto, dependendo se seu valor está mais próximo do máximo ou mínimo local, respectivamente.
- Método da Média: seleciona o limiar como a média da distribuição local de intensidade calculada no interior de uma vizinhança. A média é então subtraída de um valor constante. O limiar é calculado como

$$T = \mu(x, y) - C \quad (5)$$

em que $\mu(x, y)$ é a média em uma vizinhança local de (x, y) e C é uma constante de ajuste do limiar. Dessa forma, se o valor de um pixel (x, y) for maior do que o limiar T , o pixel será considerado como pertencente ao objeto; caso contrário, será considerado como fundo.

- Método da Mediana: seleciona o limiar como a mediana da distribuição local de intensidade. Dessa forma, se o valor de um pixel (x, y) for maior do que a mediana de sua vizinhança, o pixel será considerado como pertencente ao objeto; caso contrário, será considerado como fundo.

Para cada método de binarização, mostre os histogramas e a fração dos pixels pretos para cada imagem de entrada. Os valores de limiar e tamanhos de vizinhança utilizados nos algoritmos devem estar descritos no relatório.

2 Entrada de Dados

As imagens de entrada estão no formato PGM (*Portable GrayMap*). Alguns exemplos encontram-se disponíveis no diretório: http://www.ic.unicamp.br/~helio/imagens_pgm/

3 Especificação da Entrega

- A entrega do trabalho deve conter os seguintes itens:
 - código fonte: o arquivo final deve estar no formato *zip* ou no formato *tgz*, contendo todos os programas ou dados necessários para sua execução.
 - relatório: deve conter uma descrição dos algoritmos e das estruturas de dados, considerações adotadas na solução do problema, testes executados, eventuais limitações ou situações especiais não tratadas pelo programa.
- O trabalho deve ser submetido por meio da plataforma *Google Classroom*.
- Data de entrega: 05/05/2023.

4 Observações Gerais

- Os programas serão executados em ambiente Linux. Os formatos de entrada e saída dos dados devem ser rigorosamente respeitados pelo programa, conforme definidos anteriormente. Não serão aceitos trabalhos após a data de entrega.
- Os seguintes aspectos serão considerados na avaliação: funcionamento da implementação, clareza do código, qualidade do relatório técnico.

Referências

- [1] J. Bernsen. “Dynamic Thresholding of Grey-Level Images”, Proceedings of 6th International Conference on Pattern Recognition, Berlim, Alemanha, pp. 1251–1255, Outubro de 1986.
- [2] W. Niblack. “An Introduction to Digital Image Processing”, Prentice Hall, 1986.
- [3] N. Otsu. “A Threshold Selection Method from Gray Level Histograms”, IEEE Transactions on Systems, Man and Cybernetics, vol. 9, n. 1, pp. 62–66, Março de 1979.
- [4] N. Phansalskar, S. More, A. Sabale. “Adaptive Local Thresholding for Detection of Nuclei in Diversity Stained Cytology Images”, International Conference on Communications and Signal Processing, pp. 218–220, 2011.
- [5] J. Sauvola, M. Pietaksinen. “Adaptive Document Image Binarization”, Pattern Recognition, vol. 33 pp. 225–236, 2000.