3 Exercises

Exercise 1: Write a program to choose the threshold based on Iterative optimal threshold selection (global thresholding approach).

$$g(x,y) = \begin{cases} 1, & \text{if } f(x,y) > T \\ 0, & \text{if } f(x,y) \leq T \end{cases}$$

Algorithm 1 Global thresholding approach

procedure GLOBAL-THRESH–ALG(I is a gray image)
  Initialize threshold T (e.g T = the average gray values)
  Divide image into two groups R_1 and R_2 based on the threshold T
  Calculate the mean value μ_1 and μ_2 of R_1 group and R_2 group.
  Choose a new threshold $T = (\mu_1 + \mu_2)/2$
  Until μ_1 and μ_2 not change.
end procedure

Exercise 2: Write a program to choose the threshold based on the double thresholding for region growing.

$$g(x,y) = \begin{cases} a, & \text{if } f(x,y) > T_2 \\ b, & \text{if } T_1 < f(x,y) \leq T_2 \\ c, & \text{if } f(x,y) \leq T_1 \end{cases}$$

Algorithm 2 Double Thresholding for Region Growing

procedure DOUBLE-THRESH–ALG(I is a gray image)
  Select two threshold T_1 and T_2
  Divide image into three groups R_1, R_2 and R_3 regions; R_1 consists of the pixel values lower than T_1; $R2$ consists of the pixel values in range $T_1 \leq I(x,y) \leq T_2$; R_3 consists of the pixel values higher than T_2
  for $I(x,y) \in R_2$ **do**
    if $I(x,y)$ has a neighbor value belong to R_1 **then**
      $R_1 = R_1 \cup I(x,y)$
    end if
    Until no pixels are reassigned in R_1
  end for
  Reassign any pixels left in region R_2 to region R_3.
end procedure

Exercise 3: Write a program to segment an input image with local thresholding.

Exercise 4: Write a program to segment an input image with global thresholding approach based on Otsu's method.

Exercise 5: Write a program to segment an input image with K-mean method.

Algorithm 3 K-mean

procedure K-MEAN–ALG(I is a gray image, n clusters)
    Randomly choose k samples as $\mu_j \in \mu^{1\times K}$ the initial cluster centers $C^{1\times K} = \{C_1, C_2, .., C_k\}$
    for $\mu_j \in \mu^{1\times K}$ **do**
        Compute the distance $d(I(x,y), \mu_j)$
        Partition the pixels of image $I(x,y)$ into the clusters $\mu_j \in C_j, (j \leq K)$, which $argmax_j\{d(I(x,y), \mu_j\}$
    end for
    Update the cluster centers μ_j
    Until convergence.
    Returns the *cmap* containing the cluster indices of each pixel and V matrix containing the k cluster centroid locations.
end procedure

4 References

1. R. C. Gonzalez, R. E. Woods. Digital Image Processing. New Jersey, Prentice Hall, 2002.

2. T. Acharya. Image Processing Principles and Applications. New York, Wiley & Son, 2005

3. I.T. Young, J.J. Gerbrands, L.J. van Vliet. Fundamentals of Image Processing, Delft University of Technology, 1998.