COMP ENG 4TN4

Image Processing

Term II, January – April 2019

Project 1

Simon Ly
Student ID: 1410173
Mac ID: lysv

Background

$$T : \{0, 1, \cdots, 255\} \rightarrow \{0, 1, \cdots, 255\} \tag{1}$$

$$T(j) \geq T(i) \ \text{ if } j > i \tag{2}$$

$$T(i) = \sum_{0 \leq j \leq i} s_j, \ 0 \leq i < 256$$

$$0 \leq s_j < 256 \tag{3}$$

$$\sum_{0 \leq j < 256} s_j < 256$$

$$C(\mathrm{s}) = \sum_{0 \leq j < N_x} p_j s_j \tag{4}$$

Questions

a. Formulate the problem of contrast enhancement as a linear programming problem of maximizing the expected contrast C(s) while satisfying the monotonicity condition (2) and the range of the mapping function T in (1), and also limiting tone distortions.

From the tutorial slides, a given example shows a relaxation on constraint c by converting the real value solution s into an integer transfer function.

$$T(i) = \left\lfloor \sum_{0 \leq j \leq i} s_j + 0.5 \right\rfloor, \ 0 \leq i < L$$

Though it is not an ideal solution, the changes do not substantially change the optimality.

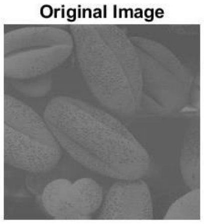
$$\max_{\mathrm{s}} \sum_{0 \leq j < L} p_j s_j$$

$$\text{subject to} \ \sum_{0 \leq j < L} s_j < \text{Ł};$$

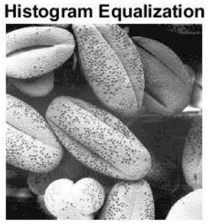
$$s_j \geq 1/d, \ 0 \leq j < L.$$

b. **Solve the above formulated linear programming problem using MatLab or any other solver, and run your program to improve the visual quality of the input images**.

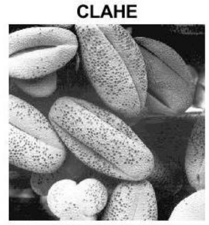
Original Image



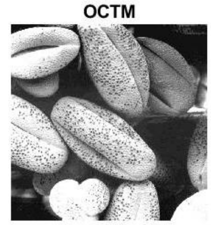
Histogram Equalization



CLAHE



OCTM



c. Write a project report to detail your algorithm development and discuss the roles of different constraints and their impacts on visual effects of the processed color image. Your conclusions and claims have to be backed up by experimental results, such as output images and plots of the input histograms and optimized transfer functions.

A linear programming approach to solving the OCTM problem is achieved by modeling the limits and boundaries in Matlab and using linprog() to initiate it's version of the dual-simplex optimization algorithm. The d value is the maximum tone distortion allowed, which is an upper bound for D(s). I set this as a lower bound because in the optimization problem it counts the number of gray levels traversed. Doing this would prevent a certain number of adjacent gray levels to be all converted to the same level. The upper bound in the case of linprog() functions in a similar concept except output to input. The b value in linprog is the output pixel value limit. In conjunction, parameters 'A' and 'b' ensures the first condition in the optimization problem is satisfied.

If the b value is changed, then the output image will be scaled accordingly to fit the limitations of the output device. This in turns interferes with the defined max distortion values. If the d is set too low then the image would look the same, if set too high it allows more distortion. If the upper bound, u, is very large, it allows the optimization algorithm to accept a step function as an appropriate solution.

Histogram equalization distributes the histogram evenly but does not consider contrasts locally. Thus we have an image with a lot more gray levels opposed to the original image but certain details are not enhanced. Contrast-limited histogram equalization slightly betters the histogram by distributing pixels with very high occurrences to the rest of the pixels but still does not fully enhance the image properly. The OCTM clearly shows a high contrast image without affecting our perception of what it is.