ECE 558 HmWK 02

Problem 1: Solve:

For any pixel in S, the set of pixels that are connected to it in S is called a connected component of. If it only has one connected component, then the set S is called a connected set or a region.

Two regions, R_i and R_j, are said to be adjacent if their union forms a connected set.

(a) Due to $V=\{1\}$,
there are 3 connected components in $S_1 \cup S_2$,
so two subsets aren't 4-adjacent.

(b) Due to $V=\{1\}$,
there is 1 connected component in $S_1 \cup S_2$,
so two subsets are 8-adjacent.

(c) Due to $V=\{1\}$,
there is 1 connected component in $S_1 \cup S_2$,
so two subsets are m-adjacent.

Problem 2: Solve:

(a) scaling and translation:

step 1: $x' = C_x \cdot x$

$y' = C_y \cdot y$

step 2: $x' = C_x \cdot x + t_x$

$y' = C_y \cdot y + t_y$

$$\xrightarrow{\text{Affine Matrix}} \begin{bmatrix} C_x & 0 & t_x \\ 0 & C_y & t_y \\ 0 & 0 & 1 \end{bmatrix}$$

(b) scaling, translation and rotation

step 1: $x' = C_x \cdot x$

$y' = C_y \cdot y$

step 2: $x' = C_x \cdot x + t_x$

$y' = C_y \cdot y + t_y$

$$\xrightarrow{\text{Affine Matrix}} \begin{bmatrix} C_x \cdot \cos\theta & -C_y \cdot \sin\theta & t_x \cos\theta - t_y \sin\theta \\ C_x \cdot \sin\theta & C_y \cdot \cos\theta & t_x \cdot \sin\theta + t_y \cdot \cos\theta \\ 0 & 0 & 1 \end{bmatrix}$$

step 3: $x' = (C_x \cdot x + t_x)\cos\theta - (C_y \cdot y + t_y)\sin\theta = C_x \cdot \cos\theta \cdot x - C_y \cdot \sin\theta \cdot y + t_x \cdot \cos\theta - t_y \cdot \sin\theta$

$y' = (C_x \cdot x + t_x)\sin\theta + (C_y \cdot y + t_y)\cos\theta = C_x \cdot \sin\theta \cdot x + C_y \cdot \cos\theta \cdot y + t_x \cdot \sin\theta + t_y \cdot \cos\theta$

(c) vertical shear, scaling, translation and rotation

step 1: $x' = x + Sv \cdot y$
$y' = y$

step 2: $x' = Cx \cdot x + Cx \cdot Sv \cdot y$
$y' = Cy \cdot y$

step 3: $x' = Cx \cdot x + Cx \cdot Sv \cdot y + tx$
$y' = Cy \cdot y + ty$

step 4: $x' = (Cx \cdot x + Cx \cdot Sv \cdot y + tx)\cos\theta - (Cy \cdot y + ty)\sin\theta = Cx \cdot \cos\theta \cdot x + (Cx \cdot Sv \cdot \cos\theta - Cy \cdot \sin\theta)y + tx \cdot \cos\theta - ty \cdot \sin\theta$

$y' = (Cx \cdot x + Cx \cdot Sv \cdot y + tx)\sin\theta + (Cy \cdot y + ty)\cos\theta = Cx \cdot \sin\theta \cdot x + (Cx \cdot Sv \cdot \sin\theta + Cy \cdot \cos\theta)y + tx \cdot \sin\theta + ty \cdot \cos\theta$

$$\xrightarrow{\text{Affine Matrix}} \begin{bmatrix} Cx \cdot \cos\theta & Cx \cdot Sv \cdot \cos\theta - Cy \cdot \sin\theta & tx \cdot \cos\theta - ty \cdot \sin\theta \\ Cx \cdot \sin\theta & Cx \cdot Sv \cdot \sin\theta + Cy\cos\theta & tx \cdot \sin\theta + ty \cdot \cos\theta \\ 0 & 0 & 1 \end{bmatrix}$$

(d) By comparing the result of (a),

I operate the transformation of translation and scaling.

step 1: $x' = x + tx$
$y' = y + ty$

step 2: $x' = Cx \cdot x + Cx \cdot tx$
$y' = Cy \cdot y + Cy \cdot ty$

$$\xrightarrow{\text{Affine Matrix}} \begin{bmatrix} Cx & 0 & Cx \cdot tx \\ 0 & Cy & Cy \cdot ty \\ 0 & 0 & 1 \end{bmatrix}$$ which is different from the result of (a)

Therefore the order of multiplication of individual matrices does make a difference!

Problem 3:

(a) Solution:

event A_i: Type I occurs for the ith image (20%)

event B_i: Type II occurs for the ith image (80%)

$P(A_1 \cap A_2 \cap A_3) = 20\% \times 20\% \times 20\% = 0.008$.

(b) $P(A_{998} \cap A_{999} \cap A_{1000}) = 20\% \times 20\% \times 20\% = 0.008$.

Therefore, the result will be the same.

Problem 4:

Solution:

(a) $g(x,y) = T[f(x,y)]$

T is an operation on f defined over a neighborhood of pixel (x,y).

Consider the smallest possible neighborhood of size 1×1

$g(x,y) = T[f(x,y)] \rightarrow s = T[r]$

For simplicity in notation, s and r are variables denoting the intensity of g and f at any point (x,y)

In order to meet the requirement,

we can derive the transformation function $s = \left[\frac{99}{256} r\right]$

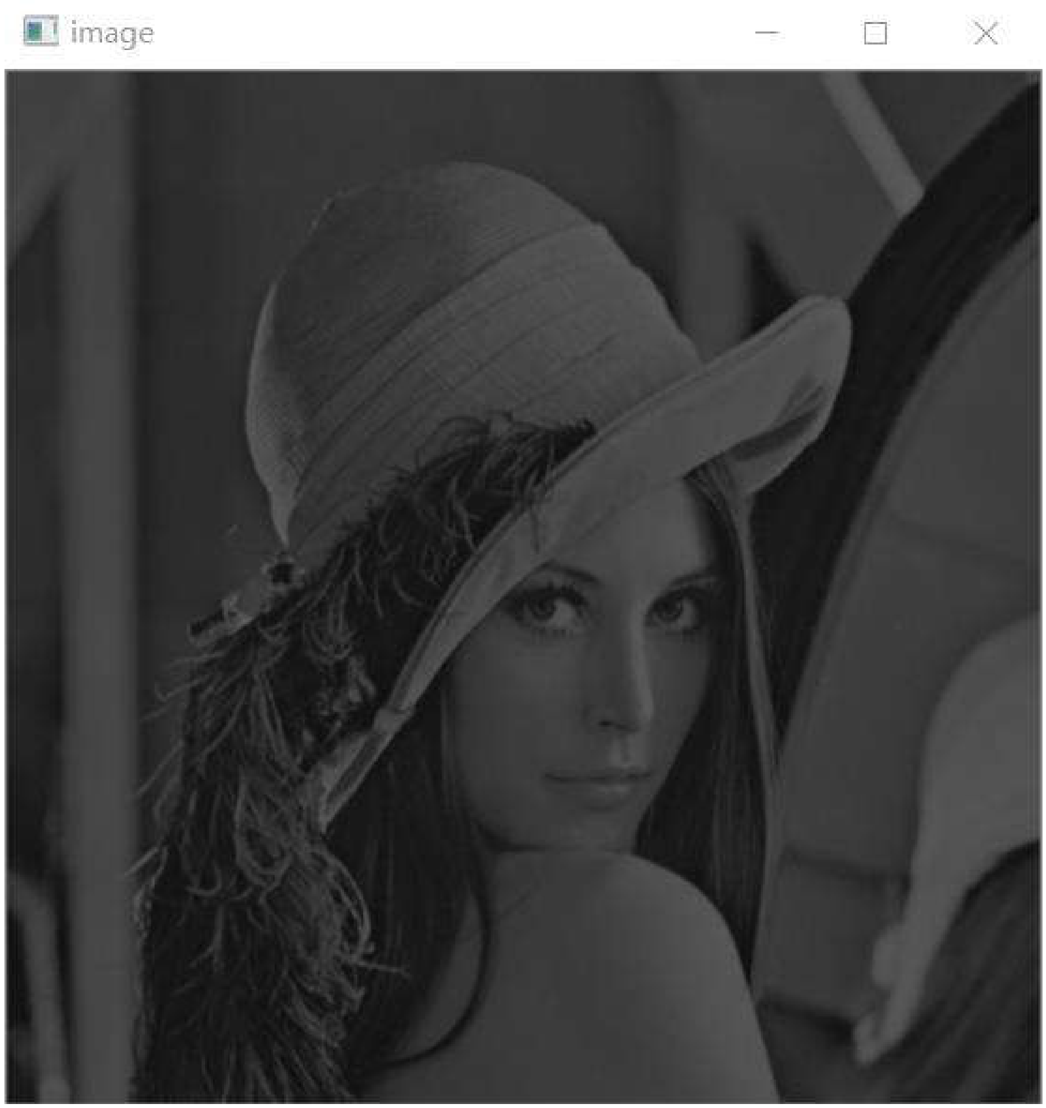
(b)

The Python code to check my function is shown as follows:

```
from pylab import *
import numpy as np
import cv2

img=cv2.imread('lena.png',0)
for m in range(0,len(img)):
    for n in range(0,len(img.T)):
        img[m,n]=round(img[m,n]*99/256)  # check my derived function
cv2.namedWindow('Image')
cv2.imshow('image',img)
cv2.waitKey(0)
cv2.destroyAllWindows()
```

The resulting image is shown as follows:



Problem 5:

Solution:

Let r be the intensity value of a random pixel in an image with intensity levels in the range $[0, 255]$

$r=0$: $n_0 = (\frac{m}{4}+\frac{m}{4})(\frac{n}{4}+\frac{n}{4}) + \frac{m}{4}(\frac{n}{4}+\frac{n}{4})\cdot\frac{1}{2} = \frac{5}{16}mn$;

$r=16$: $n_{16} = \frac{m}{4}\cdot 3\cdot\frac{n}{4} = \frac{3}{16}mn$

$r=32$: $n_{32} = \frac{m}{4}(\frac{n}{4}+\frac{n}{4})\cdot\frac{1}{2} = \frac{1}{16}mn$

$r=127$: $n_{127} = \frac{m}{4}\cdot\frac{n}{4} = \frac{1}{16}mn$

$r=191$: $n_{191} = \frac{m}{4}\cdot\frac{n}{4} = \frac{1}{16}mn$

$r=228$, $n_{228} = \frac{1}{16}mn$

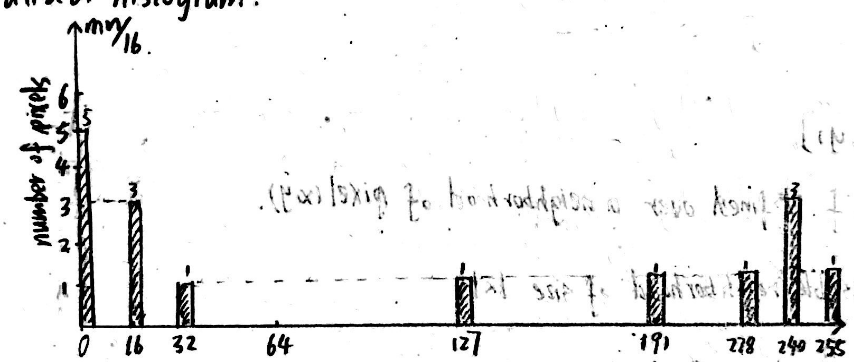
$r=240$, $n_{240} = \frac{3}{16}mn$

$r=255$, $n_{255} = \frac{1}{16}mn$.

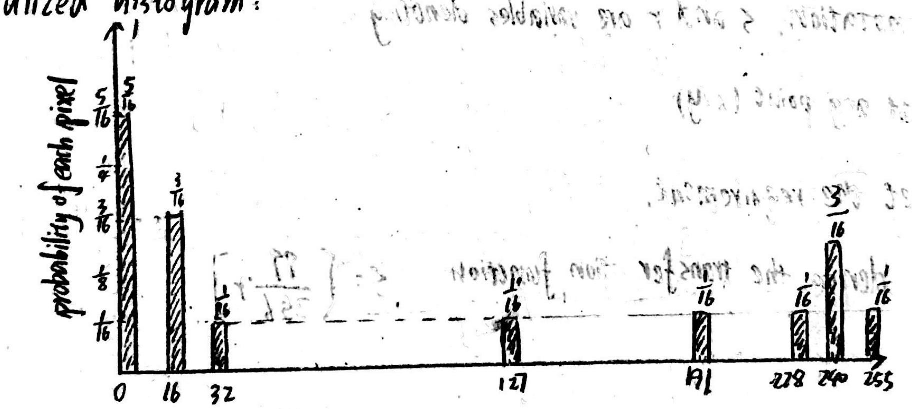
Probability mass function: $P(r=0)=\frac{5}{16}$, $P(r=16)=\frac{3}{16}$, $P(r=32)=\frac{1}{16}$

$P(r=127)=\frac{1}{16}$, $P(r=191)=\frac{1}{16}$, $P(r=228)=\frac{1}{16}$, $P(r=240)=\frac{3}{16}$, $P(r=255)=\frac{1}{16}$.

The unnormalized histogram:



The normalized histogram:



Problem 6:

Solution:

(a) $s = T(r) = (L-1)\int_0^r P_r(w)\,dw$

$$= (L-1)\int_0^r \frac{2w}{(L-1)^2}\,dw$$

$$= \frac{w^2}{L-1}\Big|_0^r = \frac{r^2}{L-1}$$

(b) $s = G(z) = (L-1)\int_0^z P_z(w)\,dw$

$$= (L-1)\int_0^z \frac{3w^2}{(L-1)^3}\,dw$$

$$= \frac{w^3}{(L-1)^2}\Big|_0^z = \frac{z^3}{(L-1)^2}$$

Therefore $z = T(s) = G^{-1}(z) = \left[(L-1)^2 \cdot s\right]^{\frac{1}{3}}$

(c) $z = T'(r) = T(s)\big|_{s=T(r)} = \left[(L-1)^2 \frac{r^2}{L-1}\right]^{\frac{1}{3}}$

$$= \left[(L-1)r^2\right]^{\frac{1}{3}}$$