Homework 3

(1) Problem statement

For each pixel $I(x,y)$,

1) Calculate quantization error

$$E(x,y)=\begin{cases} I(x,y) & \text{if } I(x,y)<128 \\ I(x,y)-255 & \text{if } I(x,y)\geq 128 \end{cases}$$

2) Spread the error according to Floyd-Steinberg

i.e.,

	(x,y)	$+\frac{7}{16}E$
$+\frac{3}{16}E$	$+\frac{5}{16}E$	$+\frac{1}{16}E$

3) Quantize new $I(x,y)$

to 0 or 255 using 128 as the threshold

(2.1) Experimental results

Original Image I (Gray Image)

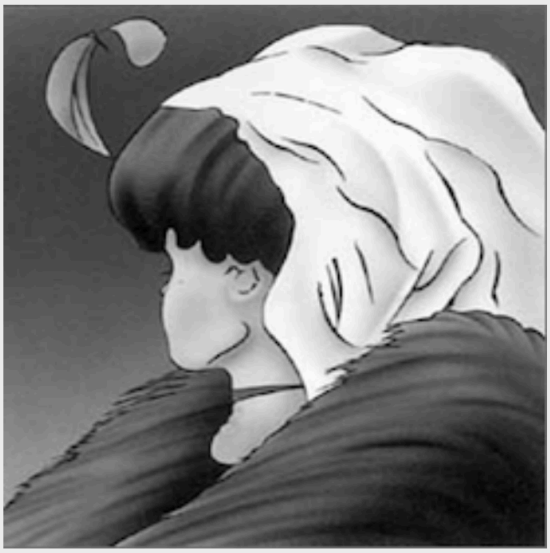
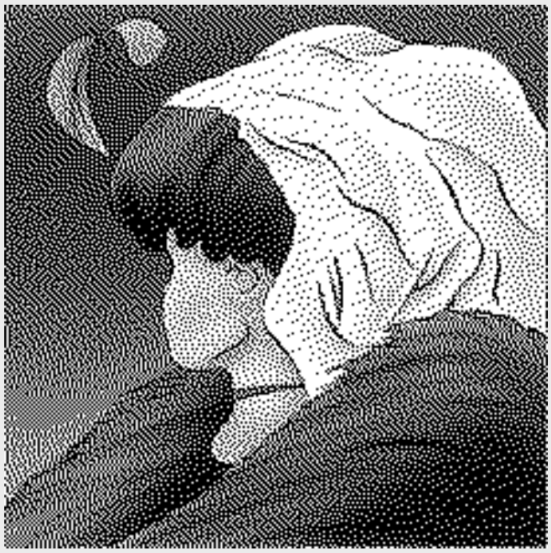


Image I' (Floyd-Steinberg error diffusion)



(2.2) Source code

```python
# HW3 (Implement Error Diffusion Dithering)
# Spread the error according to "Floyd-Steinberg"
from PIL import Image

# Input a grayscale image I
I = Image.open("grayscale.png").convert('L')
Width, Height = I.size

for y in range(Height):       # y = Height
    for x in range(Width):    # x = Width
        # Get each pixel I(x,y)
        ori_pixel = I.getpixel((x, y))

        # Step1 :Calculate Quantization Error
        if ori_pixel < 128:
            E = ori_pixel
        else:
            E = ori_pixel-255

        # Step2 : Spread the error according to "Floyd-Steinberg"
        if x < Width - 1:
            new_pixel = I.getpixel((x + 1, y)) + round(E * 7/16)
            if new_pixel > 255: new_pixel = 255
            elif new_pixel < 0: new_pixel = 0
            I.putpixel((x + 1, y), new_pixel)
        if y < Height - 1:
            new_pixel = I.getpixel((x, y + 1)) + round(E * 5/16)
            if new_pixel > 255: new_pixel = 255
            elif new_pixel < 0: new_pixel = 0
            I.putpixel((x, y + 1), new_pixel)
        if x < Width - 1 and y < Height - 1:
            new_pixel = I.getpixel((x + 1, y + 1)) + round(E * 1/16)
            if new_pixel > 255: new_pixel = 255
            elif new_pixel < 0: new_pixel = 0
            I.putpixel((x + 1, y + 1), new_pixel)
        if x > 0 and y < Height - 1:
            new_pixel = I.getpixel((x - 1, y + 1)) + round(E * 3/16)
            if new_pixel > 255 : new_pixel = 255
            elif new_pixel < 0 : new_pixel = 0
            I.putpixel((x - 1, y + 1), new_pixel)

        # Step3 : Quantize new I(x,y) to 0 or 255 (threshold)
        if ori_pixel < 128:
            I.putpixel((x, y), 0)  # threshold : 0
        else:
            I.putpixel((x, y), 255)  # threshold : 255

```

```
48  # -- Experimental results -- #
49  # Save new image I'
50  I.save("I'.png")
51  # Show images I and I'(use error diffusion dithering)
52  I = Image.open("grayscale.png")
53  I.show()
54  dithering_img = Image.open("I'.png")
55  dithering_img.show()
```

(2.3) Comments