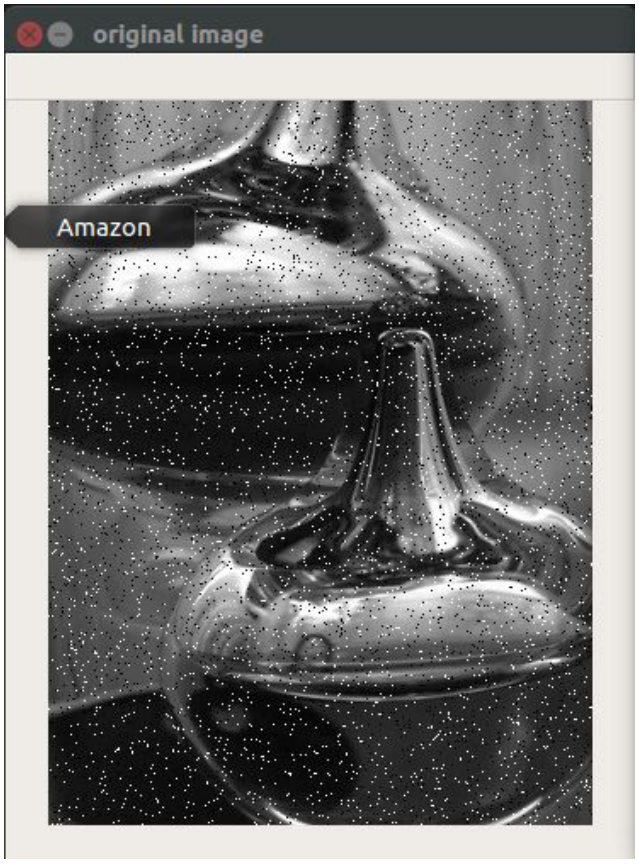
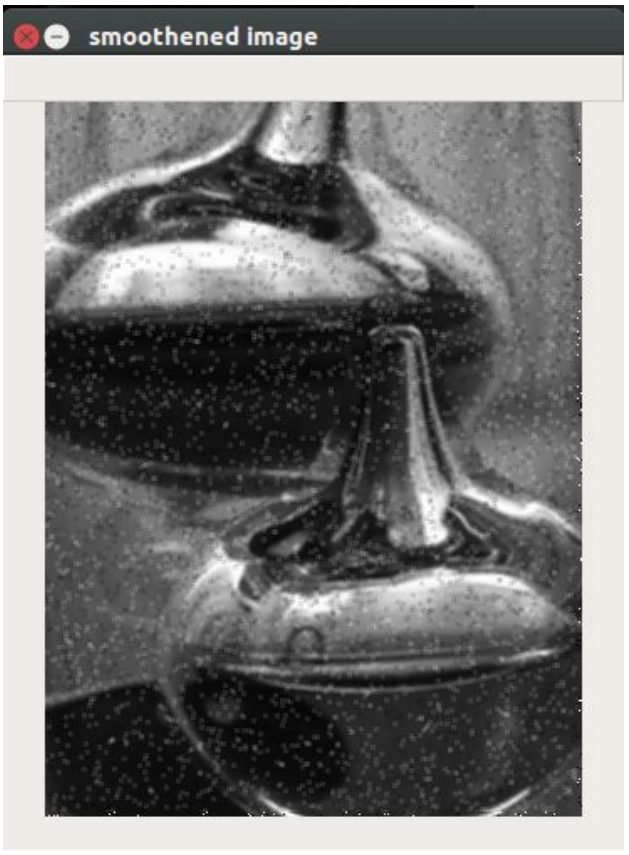
CODE:

```
import cv2 as cv
import numpy as np

src = np.array((1, 4, 6, 4, 1))
GUASSIAN_KERNEL = np.outer(src.T, src) / 256
LOW_PASS_KERNEL = np.array([1/9] * 9).reshape((3, 3))
HIGH_PASS_KERNEL = np.array([0, -0.25, 0, -0.25, 2, -0.25, 0, 0.25, 0]).reshape((3, 3))


def convolve(img1, kernel):
	img = img1.copy()
	u, v = kernel.shape
	m, n = img.shape
	for i in range(m - u):
	for j in range(n - v):
		img[i][j] = np.sum((img[i: i + u, j: j + v] * kernel))
	return img


def smoothing(img, kernel=GUASSIAN_KERNEL):
	return convolve(img, kernel)


def sharpening(img, kernel=HIGH_PASS_KERNEL):
	smooth_img = smoothing(img, LOW_PASS_KERNEL)
	return img + 2 * (img - smooth_img)


if __name__ == '__main__':
	img1 = cv.imread('img.jpg', 0)
	img2 = cv.imread('img.png', 0)

	op1 = smoothing(img1, kernel=(LOW_PASS_KERNEL))
	op2 = sharpening(img2, kernel=HIGH_PASS_KERNEL)

	'''
	cv.imshow('original image', img1)
	cv.imshow('smoothened image', op1)
	cv.waitKey(30000)
	'''
	cv.imshow('original image', img2)
	# cv.imshow('sharpened image', smoothing(sharpening(smoothing(op2))))
	cv.waitKey(30000)
```

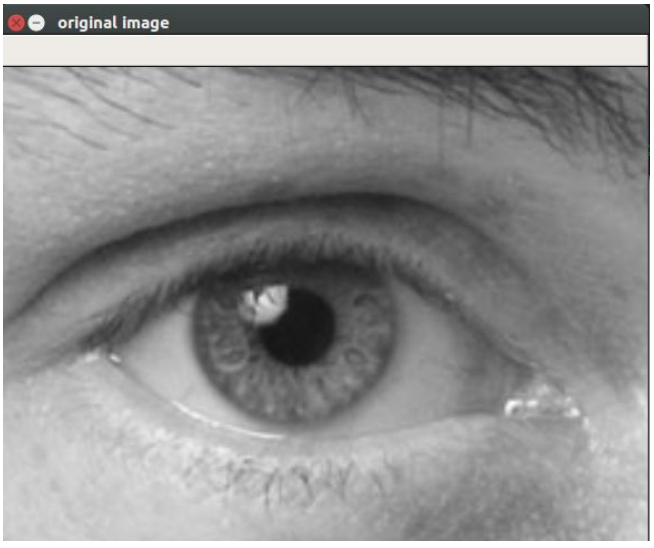
OUTPUT:



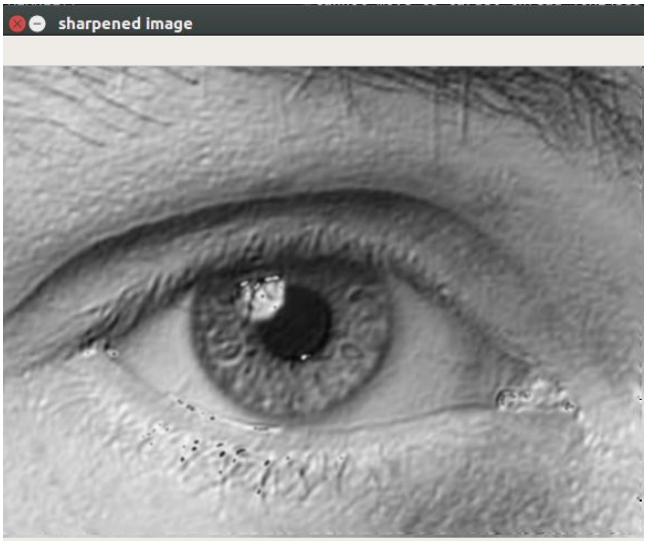
Original Image



Smoothened image



Original Image



Sharpened Image