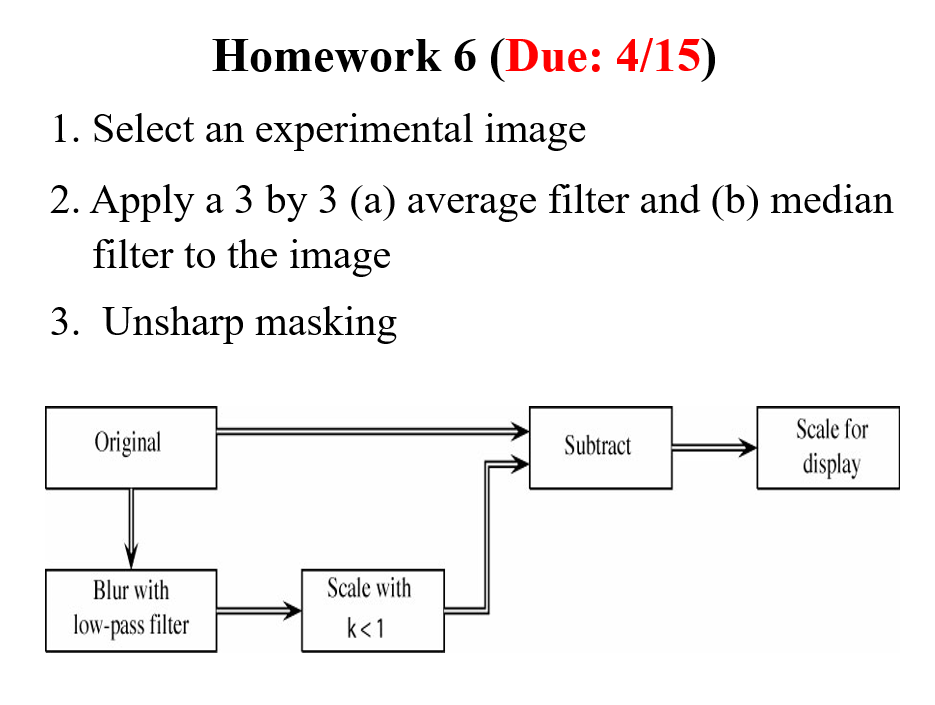
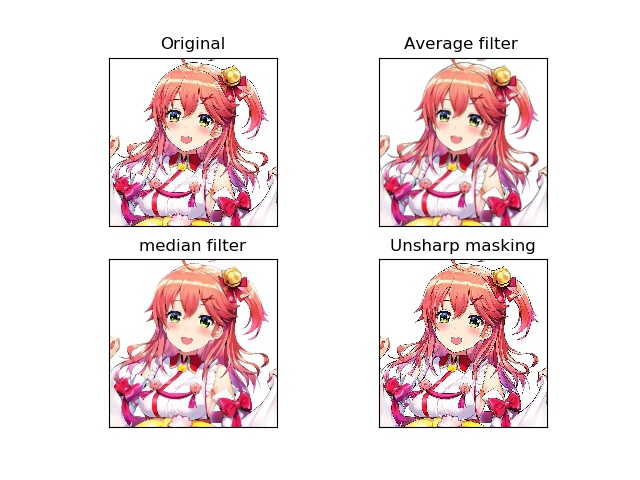
**Problem Statement：**



**Input&Output：**



**Source Code：**

**import cv2**

**import matplotlib.pyplot as plt**

**from scipy.ndimage.filters import median\_filter**

**import numpy as np**

**def unsharp(image, sigma, strength):**

**# Median filtering**

**image\_mf = median\_filter(image, sigma)**

**# Calculate the Laplacian**

**lap = cv2.Laplacian(image\_mf,cv2.CV\_64F)**

**# Calculate the sharpened image**

**sharp = image-strength\*lap**

**# Saturate the pixels in either direction**

**sharp[sharp>255] = 255**

**sharp[sharp<0] = 0**

**return sharp**

**img = plt.imread('miko.jpg')**

**sharp1 = np.zeros\_like(img)**

**for i in range(3):**

**sharp1[:,:,i] = unsharp(img[:,:,i], 5, 0.8)**

**#Average filter**

**blur = cv2.blur(img,(5,5))**

**#median filter**

**median = cv2.medianBlur(img,5)**

**plt.subplot(221),plt.imshow(img),plt.title('Original')**

**plt.xticks([]), plt.yticks([])**

**plt.subplot(222),plt.imshow(blur),plt.title('Average filter ')**

**plt.xticks([]), plt.yticks([])**

**plt.subplot(223),plt.imshow(median),plt.title('median filter')**

**plt.xticks([]), plt.yticks([])**

**plt.subplot(224),plt.imshow(sharp1),plt.title('Unsharp masking')**

**plt.xticks([]), plt.yticks([])**

**plt.savefig("miko\_filter.jpg",)**

**plt.show()**

**Comments：**

**這次的作業很簡單,這些影像處理技術在opencv都有現成的套件可以使用,不需要自己手刻,也很容易一下看出來彼此的差別及效果。在沒什麼noise的情況下,average filter和median filter看不太出來差異,在有noise的情況,median fliter 過濾雜訊的功能很好。unsharp則是會使照片輪廓變得更加明顯。**