

Project_2

408410094 資工三 葉丞勛

1.方法描述

(1)首先將柴犬飛飛的圖檔藉由`imread()`讀進來，然後利用`size()`去取得此圖的長寬以及有幾個通道，再用`im2double()`將圖檔轉成`double`的型態，值介於0~1。

(2)先用`zeros()`生成一個全為0的`matrix`，然後用三個`for loop`將本來的每個元素複製到新生成的`matrix`，在複製的時候要避免複製到邊界，確保邊界的元素全為0。

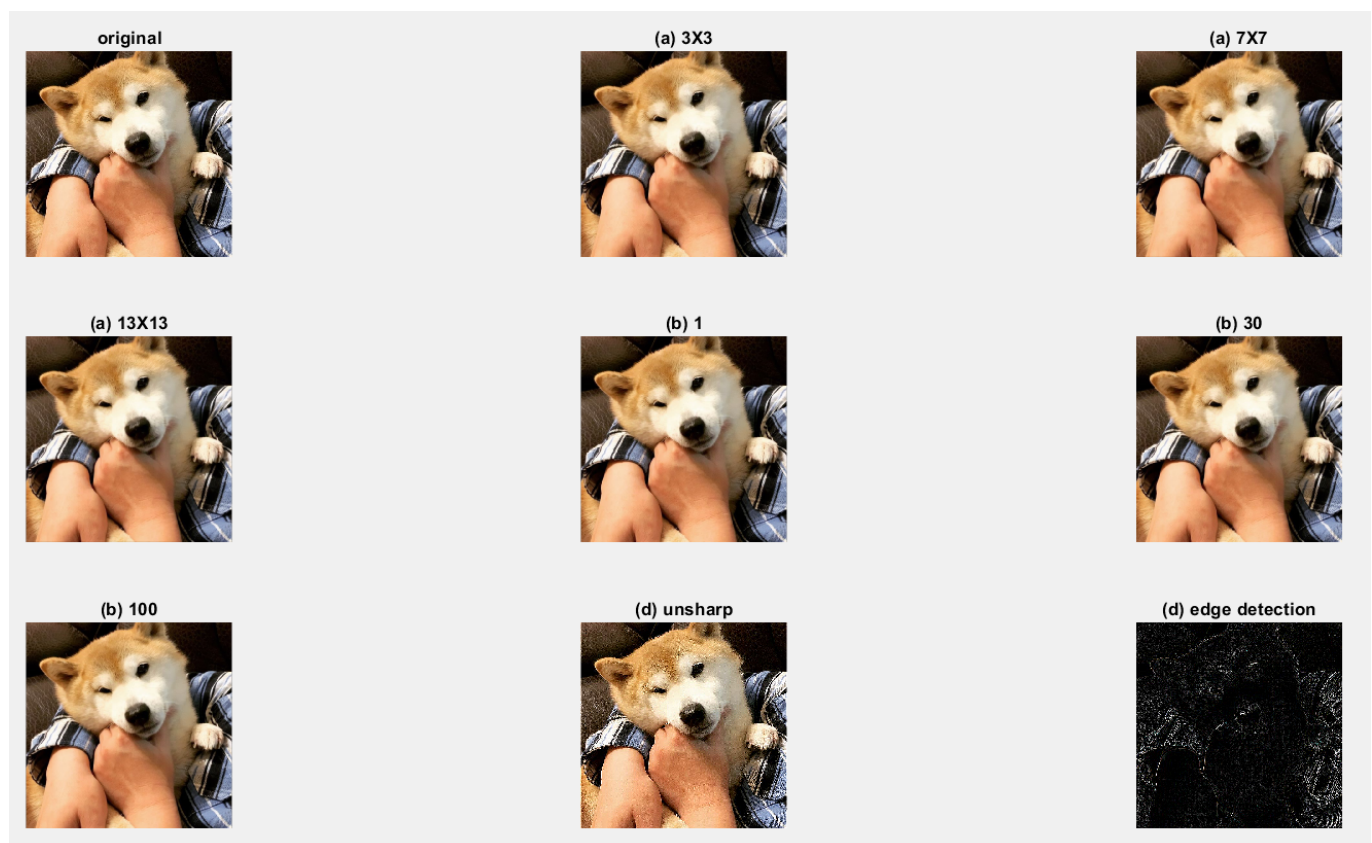
(3)用`zeros()`生成一個size跟本來圖檔完全一樣的`matrix`，再用`fspecial()`生成G，然後把G做為`convolution mask`對(2)產生的`matrix`進行`convolution`，將結果存到新生成`matrix`。

(4)用內建的`psnr()`算出PSNR值。

(5)(a)、(b)小題只要一直重複(2)~(4)即可完成，差別在於用`fspecial()`生成G時要帶進去不同的`hsize`及`sigma`而已。

(6)(d)只要調整G的值，之後的步驟都跟(a)、(b)小題一樣，做完就可以成功的將圖檔`unsharp`以及`edge detection`了。

2.實驗結果



(a)

PSNR when hsize=3X3 : 32.791791

PSNR when hsize=7X7 : 31.003117

PSNR when hsize=13X13 : 30.998945

(b)

PSNR when sigma=1 : 32.791791

PSNR when sigma=30 : 31.339621

PSNR when sigma=100 : 31.338388

3.結果討論

(a)小題:當hsize越大，代表在convolution時會考慮到越多neighboring input pixels，所以當一開始用3X3的G時只會用自己以及鄰近8個鄰居來決定output pixel，而7X7時會需要自己以及鄰近48個鄰居的資訊來決定output pixel，13X13時更需要自己以及鄰近168個鄰居的資訊來決定。當要考慮的鄰居數越多時，自己這點的值所占的比例越小，算出來的值自然會與本來的值相距越多，因此才會造成當hsize越大，誤差越大，進而導致了PSNR下降。

(b)小題:我發現當sigma越大，G矩陣中自己的權重越小，而當sigma=100時，自己與鄰居各點的權重都變得一樣。由於自己的權重越來越小，鄰居的權重越來越大的緣故，會產生當sigma越大，誤差越大的現象，進而導致造成了PSNR下降。

4.問題討論

一開始我沒有將原本的圖檔用im2double()轉成介於0~1的double型態，直接用它原本的uint8下去做，後來在做的時候發現會有overflow以及underflow的情形發生，導致unsharp以及edge detection時發生錯誤。後來先用了im2double()後就能正確將圖檔進行unsharp以及edge detection了。