

# Computer Vision Final Project Report

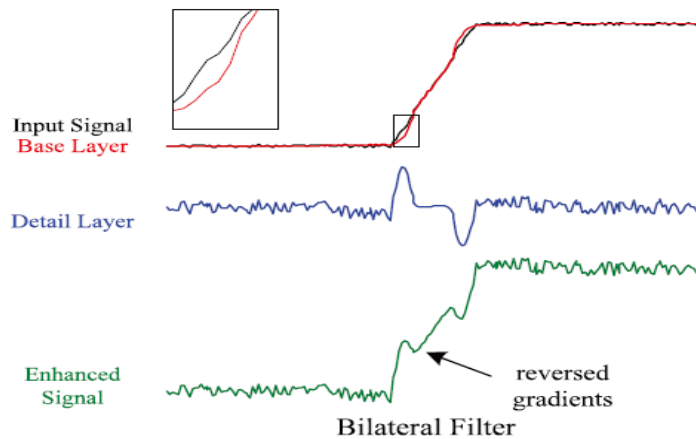
102062209 邱政凱

- 題目：An Experiment on the properties of Guided Filter

- 動機：

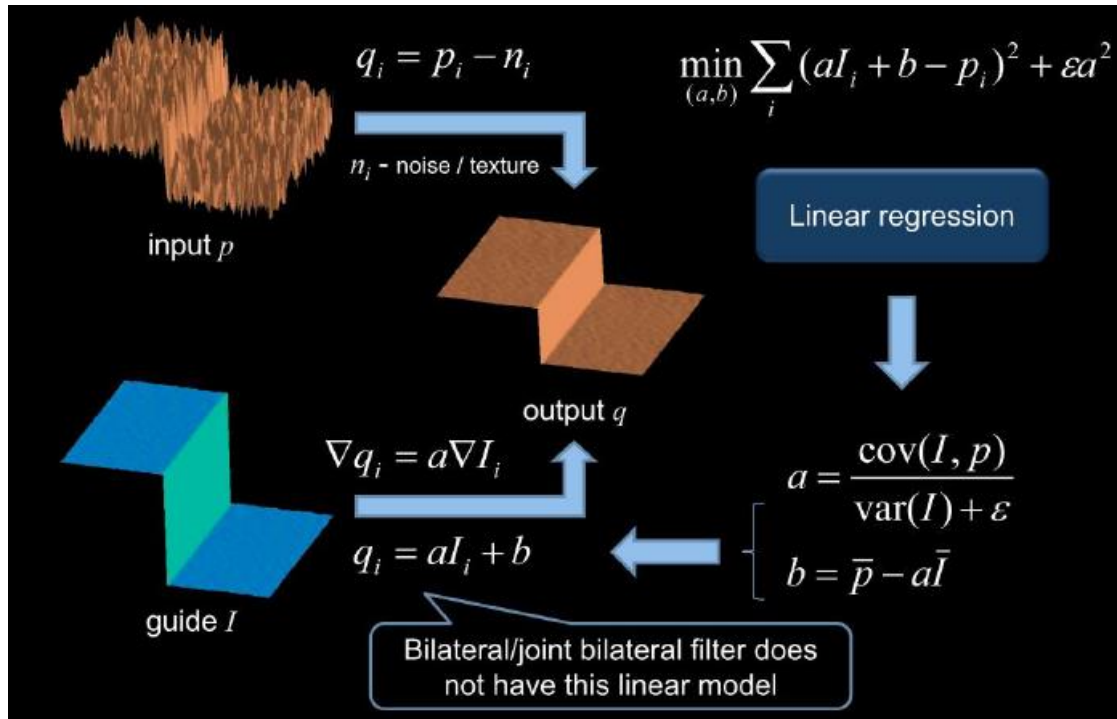
在影像處理或是電腦視覺的領域中，Smoothing 一直都是一個重要的步驟，然而若用最基本的 Gaussian Filter 去對圖片做處理，可能會讓圖片的一些細節(ex: contour)喪失。Edge-Preserving filtering 這時就有了重要的意義，可以保留住圖片中高變動的區域，藉此在保留圖片細節的狀態下去除圖片中的雜訊。

在傳統上，這項任務有幾個主要的方法來達成：Bilateral Filtering、Anisotropic Diffusion、Adaptive Smoothing...其中 Bilateral Filtering 又是其中最常被廣泛應用與研究的一個濾波器。Bilateral Filtering 的概念就是新的像素在計算的過程中除了要參考空間上鄰近的像素外，像素之間本身在色彩空間中的鄰近程度也要加入考慮。不過他有些明顯的問題，像是在沒有作加速的狀況下直接去做會需要  $O(n^2)$  的時間複雜度(因為對每個像素而言，我們都要去對 window 內所有的像素去做加權平均)、在圖片邊緣的地方會出現 Gradient Reversal 的狀況。



(Gradient Reversal，會在如 Detail Enhancement 的狀況中造成邊緣的色彩不連續。)

為了能夠發明一個更完整、功能齊全的濾波器，Microsoft Research Team 在 2010 年發表了一篇《Guided Image Filter》的論文，提出了 Guided Filter 這項技術。



- 原理:

Guided Filter 主要的原理是除了 Input Image  $p$ 、Output Image  $q$  以外，我們還需要另外一張 Guidance Image  $I$  (可以是  $p$  自己本身，這點跟 Joint Bilateral Filtering 有些相似)。

$$q_i = \sum_j W_{ij}(I) p_j,$$

$q$  的每一個像素其實都是原圖  $p$  的加權平均，而權重值則是 Guidance Image  $I$  的函數  $W$ 。Guided Filter 另一個假設就是希望  $q$  跟  $I$  之間有維持一個 Linear Model 的關係，因此可以把上式改寫為：

$$q_i = a_k I_i + b_k, \forall i \in \omega_k,$$

其中的  $a$  跟  $b$  代表的是對整張圖而言每塊 Patch 都各自擁有的參數(也因為這樣，Guided Filter 是 Patch Wise 的演算法而非 Pixel wise 的演算法，因此在後面我們可以看到 Guided Filter 可以維持住 Guidance Image 較大範圍的 Structure，因為他會保留周圍 Patch 的特性。)

而根據論文中所定義的希望最小化 Objective Function：

$$E(a_k, b_k) = \sum_{i \in \omega_k} ((a_k I_i + b_k - p_i)^2 + \epsilon a_k^2).$$

用 Linear Ridge Regression Model 可以得出  $a$  跟  $b$  如下：

$$a_k = \frac{\frac{1}{|\omega|} \sum_{i \in \omega_k} I_i p_i - \mu_k \bar{p}_k}{\sigma_k^2 + \epsilon},$$

$$b_k = \bar{p}_k - a_k \mu_k.$$

但是，一個 pixel 不應該只參考以本身為中心的 patch 中的參數，而應該是參考所有有覆蓋到該 pixel 的 patch 的像素。因此，重新定義 q 如下：

$$q_i = \frac{1}{|\omega|} \sum_{k|i \in \omega_k} (a_k I_i + b_k).$$

$$q_i = \bar{a}_i I_i + \bar{b}_i,$$

也就是許多 patch 的參數 a、b 的平均。

那 Guided Filter 為何可以有 Edge Preserving 的性質？

比較直覺的一個解釋方式是，考慮 I 跟 p 相同(Guidance Image 就是 Input Image 本身)的狀況：

$$I \equiv p. \text{ In this case, } a_k = \sigma_k^2 / (\sigma_k^2 + \epsilon)$$

$$b_k = (1 - a_k) \mu_k.$$

Case 1: "High variance." If the image  $I$  changes a lot within  $\omega_k$ , we have  $\sigma_k^2 \gg \epsilon$ , so  $a_k \approx 1$  and  $b_k \approx 0$ .

Case 2: "Flat patch." If the image  $I$  is almost constant in  $\omega_k$ , we have  $\sigma_k^2 \ll \epsilon$ , so  $a_k \approx 0$  and  $b_k \approx \mu_k$ .

至於為何可以避免掉 Gradient Reversal 的狀況？考慮  $q=aI+b$ ，對兩邊取微分，

$$\nabla q = a \nabla I.$$

而 a 依照以上定義一定是正值，也就是說 Guidance Image 跟 Output Image 的 Gradient 的 Sign 一定相同，不會有相反的狀況發生。

Guided Filter 的實作過程中，如果我們可以用 Box Filter 來實現 Integral Image 的技巧，則計算每個 patch 中所有像素的參數加總就可以很快，只要藉由幾個加減運算就可以完成，達到 O(N) 的 time complexity。

Guided Filter 的應用非常多，他可以應用在 HDR Compression、Image

Feathering(可用來做 Image Matting)、Haze Removal、flash/no flash denoising 等等非常多元的應用。

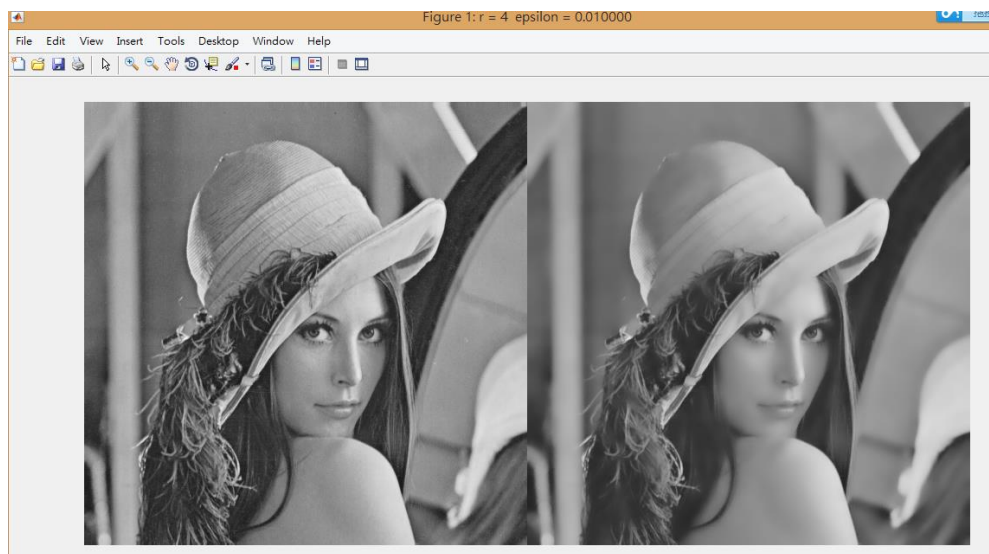
- 實作:

實作的部分主要是把論文裡面提到的原理用 Matlab 直接實作出來。實作過程基本上沒有遇到太大的困難，因為其實論文就是把每個 patch 的參數算出來，直接把論文上的公式用 matlab 打出來就好。比較特別的是為了能夠用 Integral Image 的技巧加速 Guided Filter 的執行速度，還另外寫了一個 BoxFilter 的函數算出每個像素累加的像素值，並且用點除 patch size 的方式取得每個 patch 像素值的平均值。除了一般的 Guided Filter 外，還參考了論文裡的” Extend to Color Image” 章節另外寫了一個可以用彩色圖片來當作 Guidance Image 的 GuidedFilterColor，必須計算 RGB 三個 channel 跟彼此之間的 covariance，也因此對於圖片的顏色比較敏感，不會在某些圖片的 edge 處出現 halo，或是 rgb 色彩被糊掉的狀況。

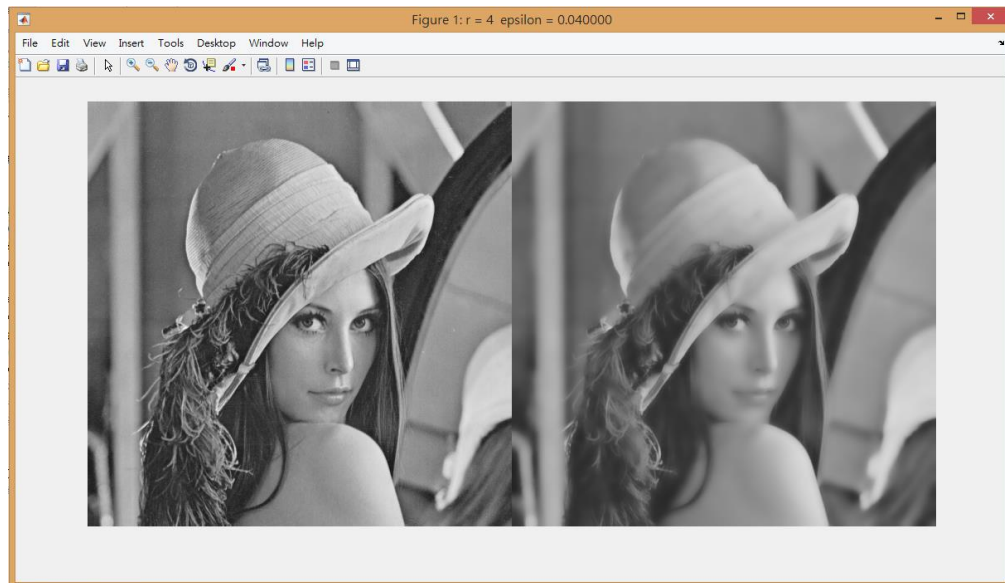
為了測試實作出來的 Guided Filter 的效果，我選了 Smoothing(使用黑白圖片跟彩色圖片當 guidance image 去比較)、Detail Enhancement、Structure Transfer、flash/no flash denoising 四個項目分別去對成果做實驗。

Guided Filter 裡主要有兩個可以控制的參數，window(patch)size  $r$ (半徑) 和 出現在 objective function  $E$  裡面的，用來 penalize 太大的  $a$  用的  $\epsilon$ 。

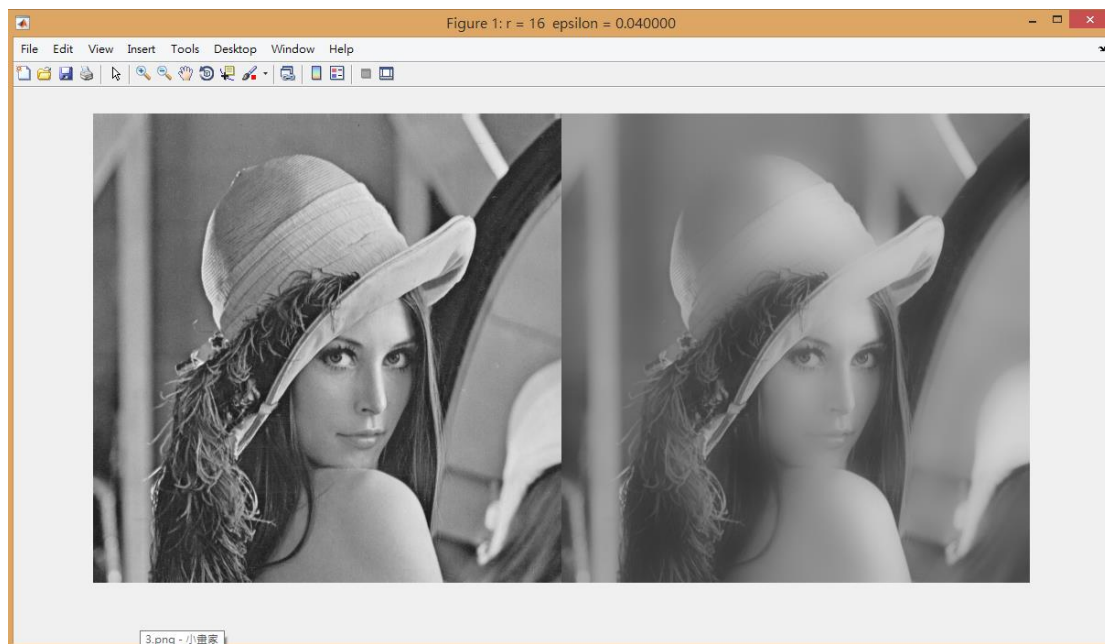
首先，用灰階 Lena 圖來測試 Edge-Preserving Smoothing 的效果：



$r=4$   $\epsilon = 0.01$



$r=4, \epsilon=0.04$

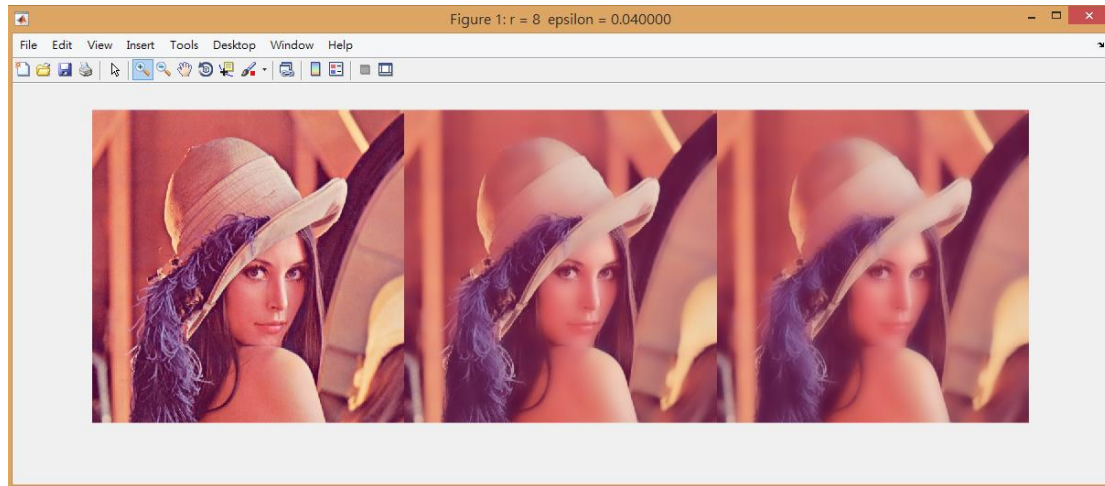


$r=16, \epsilon=0.01$

從以上的比較可以看出  $\epsilon$  越大圖片會越 smooth，不也也可能因此造成 edge preserving 的效果跑掉。不過若是增加 window 半徑，則可以更有效的達到高 smooth 的效果同時保留圖片的輪廓。(看帽子毛料的部分應該可以看的出來差別)

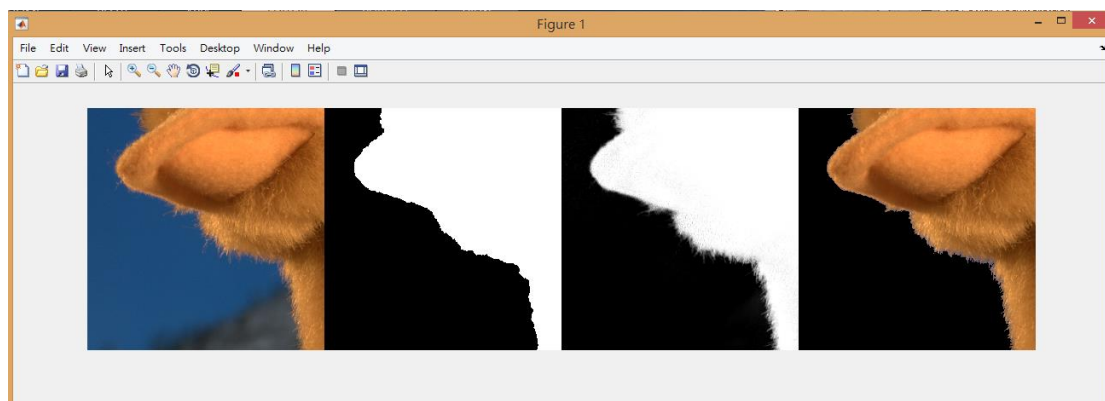
接下來是測試對同樣的一張 RGB 圖，若用同樣的圖 Guidance image，但是分為灰階和彩色來分別 Guide 會有什麼差別。





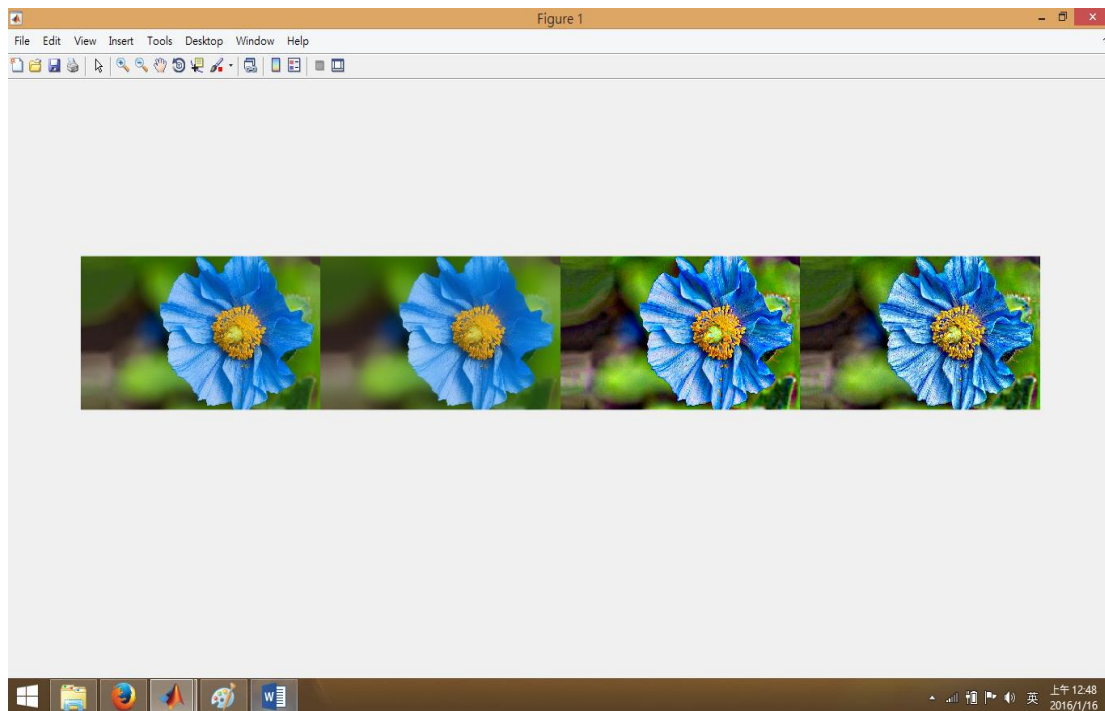
左邊是原本的彩色 Lena 圖，中間則是用彩色的 Lena 圖自己當 Guidance Image，右邊則是用灰階的 Lena 圖來當 Guidance Image。可以看出若原圖是彩色的，則若單純用灰階的圖片去 Guide，可能會造成沒有辦法把輪廓保留得很好，但是若是用彩色的圖來 Guide，則可以很好的保留原圖的輪廓。

接下來試驗 Structure Transfer 的效果。



可以看到，如果我用左邊的圖當 Guidance Image，那麼我只需要給予一張大致形狀的 binary mask，Guided Filter 就可以把左邊的圖片細節 transfer 到我們的 mask 上變成第三張圖，如果再用這個 structure transfer 後的 binary mask 去對左邊的圖做 mask，就可以達到 Image Matting 的效果(最右邊)。

接下來試驗 Detail Enhancement 的效果，



最左邊是原圖，第二張是用 Guided Filter smooth 過後的 Base Layer。右邊兩張是把原圖-base layer 得到的 detail layer 加強後加回原圖去得到的 Enhanced 版，第三張用的是 Guided Filter，第四張用的則是 Bilateral Filter。

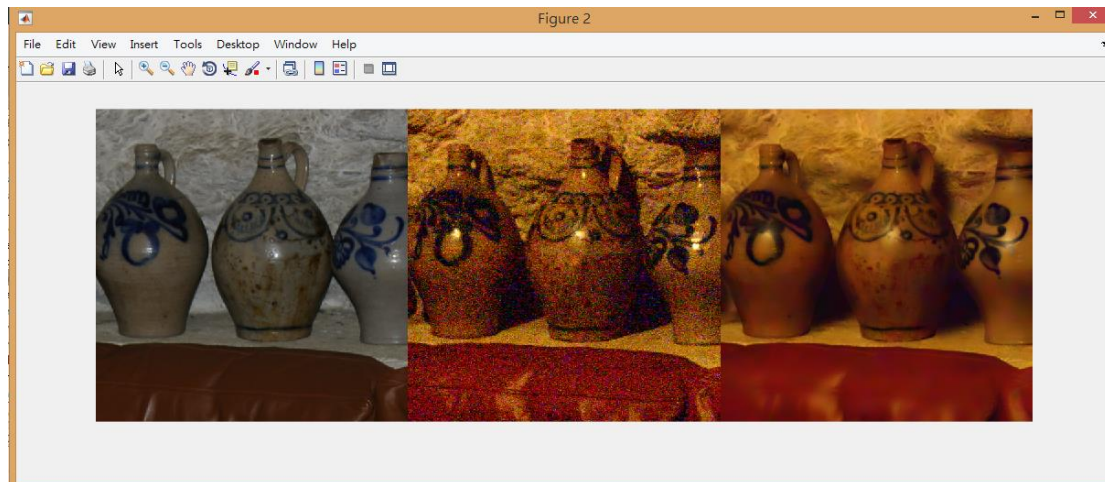


這是用 Bilateral Filtering 做的細節，可以發現在邊緣的藍線跑了出去，中間夾了背景的綠色，這就是 Gradient Reversal。



而這張則是用 Guided Filter 做的細節，可以發現 Gradient Reversal 的狀況被改善了。

最後則是測試 flash/no flash denoising。



中間是我們充滿雜訊的原圖，左邊則是用閃光燈拍照來去除雜訊後的圖。不過因為用了閃光燈的關係原本環境光的色調整個被改變了。然而如果我們用左邊的有閃光燈的圖當 Guidance Image 去對充滿雜訊的原圖做 Guided Filter 則可以得到右圖，也就是維持原圖環境光的狀態下去除雜訊的圖。

Guided Filter 是個概念簡單卻相當強大的一個濾波器，他的用途實在是非常多元，沒有時間把論文裡面有提到的應用都做過一遍實在是有點可惜。