

**原圖 lenna-RGB.tif**



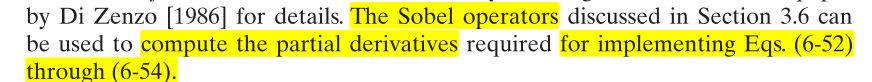
|  |  |  |
| --- | --- | --- |
|  |  |  |

**use a gradient computed in RGB color vector space to detect the edge for the image**

方法: 參考 課本p.451, p.452計算gradients(如下):

一張含有 桌 的圖片

自動產生的描述



Eqs (6-52~6-54) 在實作尚可用 Sobelx, Sobely 實現

一張含有 文字 的圖片

自動產生的描述

最後所產生的F(x, y)，即是用**gradient computed in RGB color vector space**所產生出的edge detection圖，結果如下頁。

一張含有 文字 的圖片

自動產生的描述

一張含有 文字 的圖片

自動產生的描述

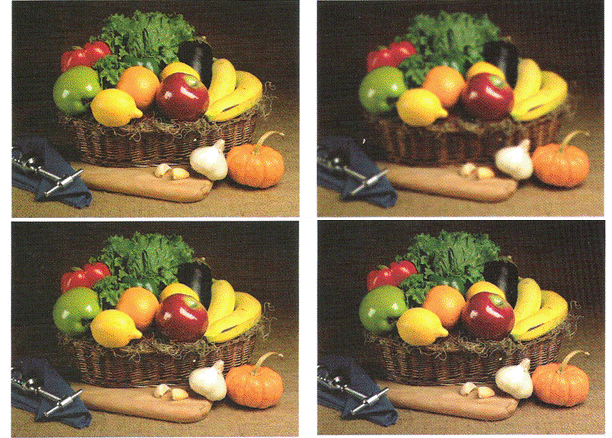
一張含有 文字 的圖片

自動產生的描述



**原圖 Visual resolution.gif**

(這邊要特別注意原本拿到的圖的shape是(438, 612, 4), 第四個維度是一條水平的黑線，應該是誤植了，因此實際在做取前三個channel即可)



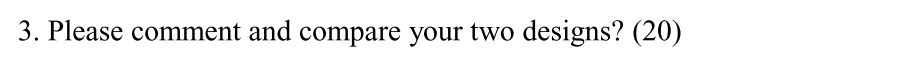
|  |
| --- |
|  |
|  |
|  |

比較R, G, B三個 channel 的影像，可以發現B channel影像偏暗，這也合理，因為原圖幾乎沒有藍色(或由藍色混和成)的地方。  
**use a gradient computed in RGB color vector space to detect the edge for the image**

方法: 同樣參考 課本p.451, p.452計算gradients

**公式及程式碼同上題**





這邊想討論兩個問題:

1. np.arctan vs. np.arctan2

2. 比較使用gradient computed in RGB color vector space 與Sobel 做edge detection的差異

**1. np.arctan vs. np.arctan2**

在python套件裡有兩個可計算 arctan的套件，兩者return的角度範圍不同:

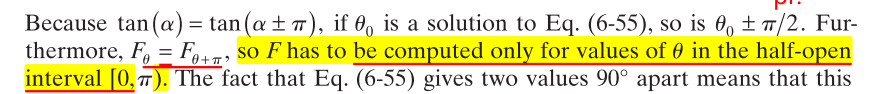
**np.arctan2 : The output of the function range from -180 to +180**

**np.arctan : The output of the function range from -90 to +90**

主要是np.arctan2可return在四個象限中的角度, 而np.arctan僅能return 兩個象限中的角度

Reference : <https://geo.libretexts.org/Courses/University_of_California_Davis/GEL_056%3A_Introduction_to_Geophysics/Geophysics_is_everywhere_in_geology.../zz%3A_Back_Matter/Arctan_vs_Arctan2>

而這邊我根據 課本p.452 內容，**在實作上使用np.arctan2，並取abs**





**2. 比較使用gradient computed in RGB color vector space 與Sobel 做edge detection的差異**

(以第一小題為例)

這邊試著做出課本 p.453 FIGURE 6.44(c) 的圖，依課本做法，分別在R, G, B 3 channel 做Sobel，再把三者相加起來。(Sobel的實作與先前HW3\_4的作法相同)

一張含有 文字 的圖片

自動產生的描述

一張含有 文字 的圖片

自動產生的描述

|  |
| --- |
| 一張含有 文字 的圖片  自動產生的描述  一張含有 文字, 書 的圖片  自動產生的描述一張含有 文字, 書 的圖片  自動產生的描述 |

用 Sobel 做出來的結果

一張含有 文字, 書 的圖片

自動產生的描述

**比較使用gradient computed in RGB color vector space 與Sobel 做edge detection的差異**

|  |  |  |
| --- | --- | --- |
| 一張含有 文字 的圖片  自動產生的描述 | 一張含有 文字, 書 的圖片  自動產生的描述 |  |

其實相較課本結果兩個方法做出來都比課本差，但edge detection的效果還是有出來，不過這邊做出來的結論與課本不同: Sobel做出來的edge detection較細緻(也有將差異值max, min print出如下以證明)，這邊把一些差異用紅色圈出。



