# Computer Vision HW1 Report

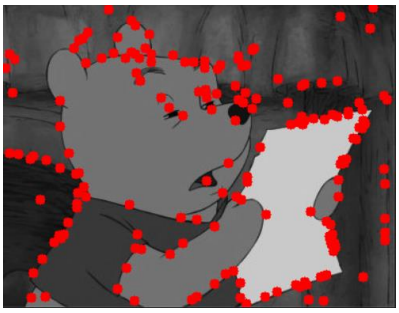
Student ID: R11921100
Name: 溫威領

## Part 1.

- **Visualize the DoG images of 1.png.**

| | DoG Image (threshold = 3) | | DoG Image (threshold = 3) |
|---|---|---|---|
| DoG1-1.png |  | DoG2-1.png |  |
| DoG1-2.png |  | DoG2-2.png |  |
| DoG1-3.png |  | DoG2-3.png |  |
| DoG1-4.png |  | DoG2-4.png |  |

- **Use three thresholds (1,2,3) on 2.png and describe the difference.**

| Threshold | Image with detected keypoints on 2.png |
|---|---|
| 1 |  |
| 2 |  |
| 3 |  |

(describe the difference)

Keypoints 大多產生在圖片物件的邊緣。Threshold 較小時在背景區域有產生 keypoint，Threshold 越大則背景的 keypoint 減少，大多落在前方角色物件中的邊緣，代表背景的 keypoints 數值較低被 Threshold 過濾掉而消失。

## Part 2.

- **Report the cost for each filtered image.**

| Gray Scale Setting | Cost (1.png) |
|---|---|
| cv2.COLOR_BGR2GRAY | 38009491 |
| R*0.0+G*0.0+B*1.0 | 43206294 |
| R*0.0+G*1.0+B*0.0 | 36887563 |
| R*0.1+G*0.0+B*0.9 | 42809236 |
| R*0.1+G*0.4+B*0.5 | 40169007 |
| R*0.8+G*0.2+B*0.0 | 38043015 |

| Gray Scale Setting | Cost (2.png) |
|---|---|
| cv2.COLOR_BGR2GRAY | 8544452 |
| R*0.1+G*0.0+B*0.9 | 5937722 |
| R*0.2+G*0.0+B*0.8 | 6510745 |
| R*0.2+G*0.8+B*0.0 | 8403523 |
| R*0.4+G*0.0+B*0.6 | 6570421 |
| R*1.0+G*0.0+B*0.0 | 6609698 |

- **Show original RGB image / two filtered RGB images and two grayscale images with highest and lowest cost.**

| Original RGB image (1.png) | Filtered RGB image and Grayscale image of Highest cost = 43206294 (r, g, b = 0, 0, 1) | Filtered RGB image and Grayscale image of Lowest cost = 36887563 (r, g, b = 0, 1, 0) |
|---|---|---|
|  |  |  |
| |  |  |

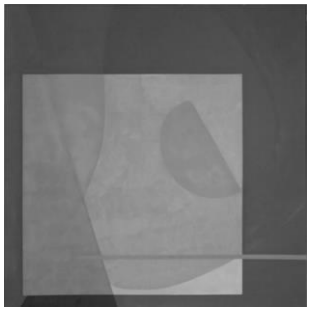
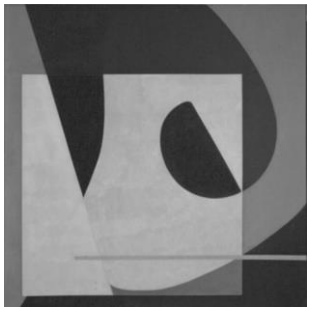
(Describe the difference between those two grayscale images)

左圖(r, g, b = 0, 0, 1)的 cost 較高，其灰階圖片整張偏暗。紅色葉子與綠色草地轉成灰階的數值都很低且差異並不大，難以辨認。

右圖(r, g, b = 0, 1, 0)的 cost 較低，其灰階圖片亮暗有明顯區別，紅色葉子的灰階數值較低，綠色草地灰階值較高，易於辨認。

從上述觀察得知，cost 較低，彩色圖片轉成灰階圖片時較易於辨認不同的顏色區塊。

單從 rgb 的觀察，因為彩色圖片中大部分是紅色和綠色元素，因此主要取藍色(左，r, g, b = 0, 0, 1)時，紅色和綠色會不易分辨，相反的如果主要取綠色(右，r, g, b = 0, 1, 0)時，則會易於辨認(綠色較亮，其他較暗)。

| Original RGB image (2.png) | Filtered RGB image and Grayscale image of Highest cost = 8544452 (r, g, b = cv2.COLOR_BGR2GRAY) | Filtered RGB image and Grayscale image of Lowest cost = 5937722 (r, g, b = 0.1, 0.0, 0.9) |
|---|---|---|
|  |  |  |
| |  |  |

(Describe the difference between those two grayscale images)

左圖(r, g, b = cv2.COLOR_BGR2GRAY)的 cost 較高，圖片中不同色塊轉成灰階的數值後，看起來分成兩區色塊(外框和內框)，其中兩區塊內顏色差異並不大，難以辨認。

右圖(r, g, b = 0.1, 0.0, 0.9)的 cost 較低，其灰階圖片不同色塊有較明顯區別，易於辨認。

從上述觀察得知，cost 較低，彩色圖片轉成灰階圖片時較易於辨認不同的顏色區塊，同 1.png。

- **Describe how to speed up the implementation of bilateral filter.**

1. 使用 numpy 函式，如 np.roll()、np.arange()進行運算盡量取代使用 for loop。
2. 建立 Gr 和 Gs 的 look up table 避免重複計算，以降低計算量與計算時間。
3. 參考 https://github.com/Spheluo/Joint-Bilateral-Filter/blob/main/JBF.py 之技巧提高速度。