

Report

R06943148 謝孟洋

1. Input images:

1a.png , 1b.png , 1c.png



2. How to run:

`python hw1.py [-h] [-i INPUT_DIR] [-o OUTPUT_DIR] [-v]`

- a) 可以用 `-i` 指定 **test images** 放在哪個資料夾，預設為 `./testdata`
因為我的學號餘數是 1，所以程式會自動到 `INPUT_DIR` 裡面找所有 1 開頭的檔案進行後續處理。
- b) 可以用 `-o` 指定 **output images** 放在哪個資料夾，預設為 `./output_imgs`。若當前的資料夾內沒有名為 `OUTPUT_DIR` 的資料夾，程式會自動建立；若當前的資料夾內存在名為 `OUTPUT_DIR` 的資料夾，程式會先將裡面的檔案全部刪除。
- c) 如果輸入 `-v`，程式在執行時會輸出更細節的資訊，包括每次 **joint bilateral filter** 所花的時間、每次競賽投票的結果等等。

3. Local minima selection:

Iterate 每一個點 (wb, wg, wr) ，如果與此點相鄰的所有其他點的 **cost** 都比此點高，則此點為一個 **local minima**。

所謂相鄰的定義為固定其中一個 **weight** 後，另外兩個 **weight** 加一或減一的組合 (須保持 3 個 **weight** 相加 = 1)。

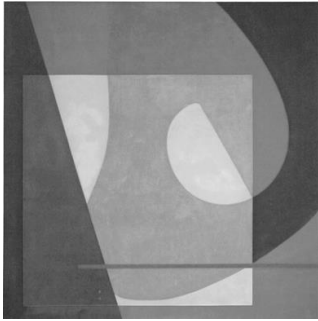
ex:

(wb, wg, wr) 相鄰的点為

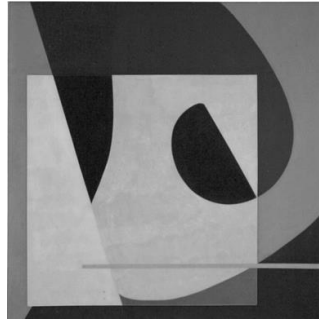
$(wb, wg+0.1, wr-0.1)$. $(wb, wg-0.1, wr+0.1)$. $(wb+0.1, wg, wr-0.1)$.
 $(wb-0.1, wg, wr+0.1)$. $(wb+0.1, wg-0.1, wr)$. $(wb-0.1, wg+0.1, wr)$

4. Result:

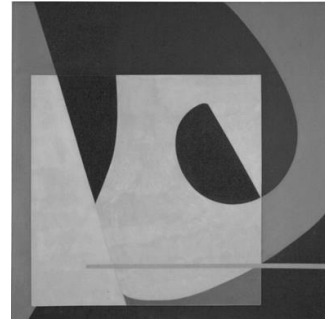
格式：檔名 (wb, wg, wr)



1a_y1 (0.0 , 0.0 , 1.0)



1a_y2 (1.0 , 0.0 , 0.0)



1a_y3 (0.9 , 0.0 , 0.1)



1b_y1 (0.0 , 0.0 , 1.0)



1b_y2 (0.0 , 1.0 , 0.0)



1b_y3 (0.5 , 0.5 , 0.0)



1c_y1 (0.0 , 0.0 , 1.0)



1c_y2 (0.0 , 0.3 , 0.7)



1c_y3 (0.0 , 0.9 , 0.1)