

Computer Vision HW1 - Advanced Color-to-Gray Conversion

電子所博一 楊仲萱 F07943023

A. Design the joint bilateral filter

主要實現 joint bilateral filter 的流程圖在下圖 figure.1 中可以看見。

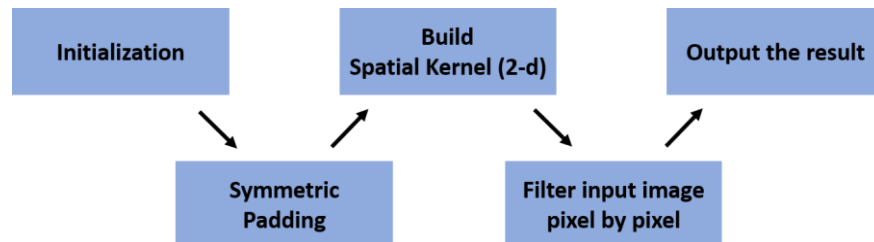


Figure.1

a-1. Initialization

首先在 Initialization 時，會知道現在的 input 圖片的長寬為何，以及根據外面所定義的 σ_s 知道現在的 window size 與對應的 radius 為何，來完成初始化。

a-2. Symmetric Padding

因為在之後使用 window 滑過每一個 pixel 時，邊界的點會需要超出原本 input image 的邊界點以使得公式也可以照常被使用。Guidance 的部分多需要注意的是要根據進來的圖片是二維還是三維的來有不同大小的 padding，都是在前兩維的邊界上上下下各都增加 radius 的長度，三維部分只是讓第三維 channel 的數量依然要保持在 3。

a-3. Spatial Kernel

Spatial Kernel 對於 pixel 實際值是多少對其本身是沒有影響的，只根據與中心要算的 pixel 距離為何來決定，因此我在進入 for loop 之前，先創好一個二維的 (window_size, window_size) 矩陣，裡面的值就是固定的 spatial kernel 的 value，在之後掃過每一個 pixel 時就不需要重新再計算此 spatial kernel。

a-4. Filter the input image pixel by pixel

由於 range kernel 會因為目前要計算的中心點 pixel 是在哪裡而有變化，因此 range kernel 的部分就是放在以 input image 的長寬所形成的 for loop 中。在這個 for loop 裡，我會先截出要被使用的 input image (window_size*window_size)，再拿 guidance image 將對應的 range kernel 計算出來並且與先前計算好的 spatial kernel 相乘。最後和 sliced input image 相乘，並作最後的 normalization，完成此 image 經過 joint bilateral filter。

a-5. Output the result

最後我使用和 input image 相同大小的 array 拿承接前面步驟後所算出的 pixel 值，並 return 此 array 回去。

B. Implement the local minima selection

Index	X	Y	Z	Cost	Vote
0	0	0	1	95074	2
1	0.1	0	0.9	76455	0
...
65	0	1	0	109948	7


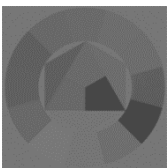
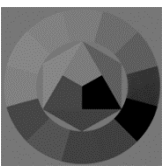

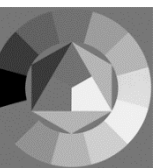
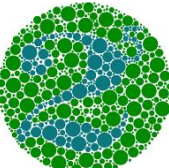
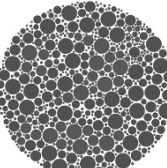
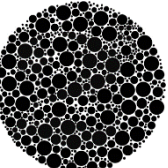
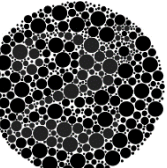
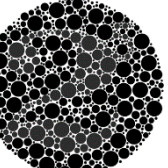





Figure.2

在投票的過程中，我主要是利用如同 figure.2 中顯示的二維矩陣，來儲存這 66 種不同的 weight 組合，順序是在 $x+y+z = 1$ 的平面上形成的三角形，由上到下，由左到右把 66 個 x, y, z 的可能都先存在此 array 中。

在同一種 sigma_r 與 sigma_s 的情況下，對這 66 種 weight 所做出來的灰階圖片當作 joint bilateral filter 的 guidance 圖片，並計算他和原本的 bilateral filter 之間每個 pixel 總共差了多少的值，並把此 cost 存在第四欄內，當所有 weight 都將 cost 計算好後，就在跑過一次所有 66 個點，比較和其旁邊的 cost 是否為 local minima，如果是的話就在 vote 那一欄那加上一。

最後再跑完所有 9 種的 sigma 組合後，將此 array 的最後一欄 print 出來，即可以知道哪一種 weight 組合得到最多的票數。

C. Input and output Image and the corresponding weight

Original	Conventional	Rank 1	Rank 2	Rank3
		 Votes: 9 (0.0,1.0,0.0)	 Votes: 7 (0.0,0.0,0.1)	 Votes: 7 (1.0,0.0,0.0)
		 Votes: 6 (1.0,0.0,0.0)	 Votes: 1 (0.8,0.0,0.2)	 Votes: 1 (0.7,0.0,0.3)
		 Votes: 7 (1.0,0.0,0.0)	 Votes: 2 (0.9,0.1,0)	 Votes: 1 (0.0,0.0,1.0)