2017 CV Homework01 Report

資科碩一 / 0556087 / 謝禎原

Outline ：

- 程式流程
- 執行方式
- 疑惑
- 執行結果

程式流程 ： (深褐色表資料結構)

Step 1 . 讀入光源

LightSources - 6(圖片數) x 3 的 Mat .

Step 2. 讀入圖片

使用 imread([path], IMREAD_GRAYSCALE) .
Images[6] - 圖片灰階值的 Mat 陣列 .

Step 3. 消除圖片中雜訊(針對 special 資料夾內的圖)

九宮格內的點的灰階值先進行排序，然後取中位數當作該 Pixel 的值 .

Step 4. 將所有圖片的灰階值整理成一個大矩陣

allImagesIntensity –
6(圖片數) x Pixel Amount(圖片長*寬) 的 Mat .

Step 5. 計算各 pixel 的 normal

normals – 3 x pixelAmout 的 Mat .

Step 6. Normalize normal 們 (但後來發現其實不用)

Step 7. 計算 df/dx 以及 df/dy

df0fdx/df0fdy – imageHeight x imageWidth 的 Mat .

Step 8. 積出深度值 Z

方法一 :

先積 X,再積 Y,再分成 Y 由小積向大或大積向小的 depth 值 .

depthXFirst –

imageHeight x imageWidth 的 Mat , 小積向大的 depth .

depthX2First –

imageHeight x imageWidth 的 Mat , 小積向大的 depth .

最後依 path 的遠近來混合以上兩者,得出真正該 pixel depth .

方法二 :

同方法一, 只是改成先積 Y 再積 X .

depthYFirst –

imageHeight x imageWidth 的 Mat , 小積向大的 depth .

depthY2First –

imageHeight x imageWidth 的 Mat , 小積向大的 depth .

方法三 **(最終使用方法)** :

也就是合併方法一和二, 依 path 的遠近來混合以上四者算出來的 depth 來作為真正的 depth .

Step 9. 平滑化

將九宮格的點的 depth 值取平均當作中心點的 depth 值, 也就是最終的 depth 了 !

depth – imageHeight x imageWidth 的 Mat , 最終的 depth .

執行方式 ：

整個專案為 **Xcode** 專案，直接開啟**.xcodeproj** 並執行，或是將**.cpp** 利用 **Visual Studio** 執行 .

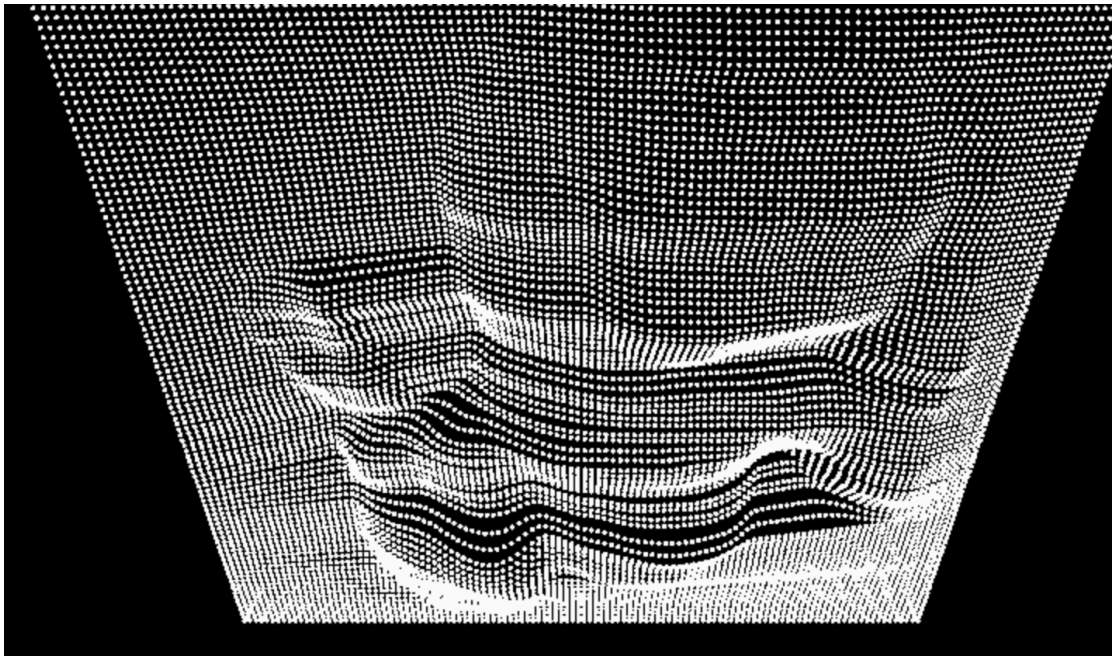
疑惑 ：

在使用方法三時發現一個問題，就是 **df/dy** 和 **df/dx** 對 **Z** 正的方向似乎相反，因此若同時考慮 **X,Y** 方向積分混合時就會出現抵銷狀況，變成一個蠻平穩的圖.

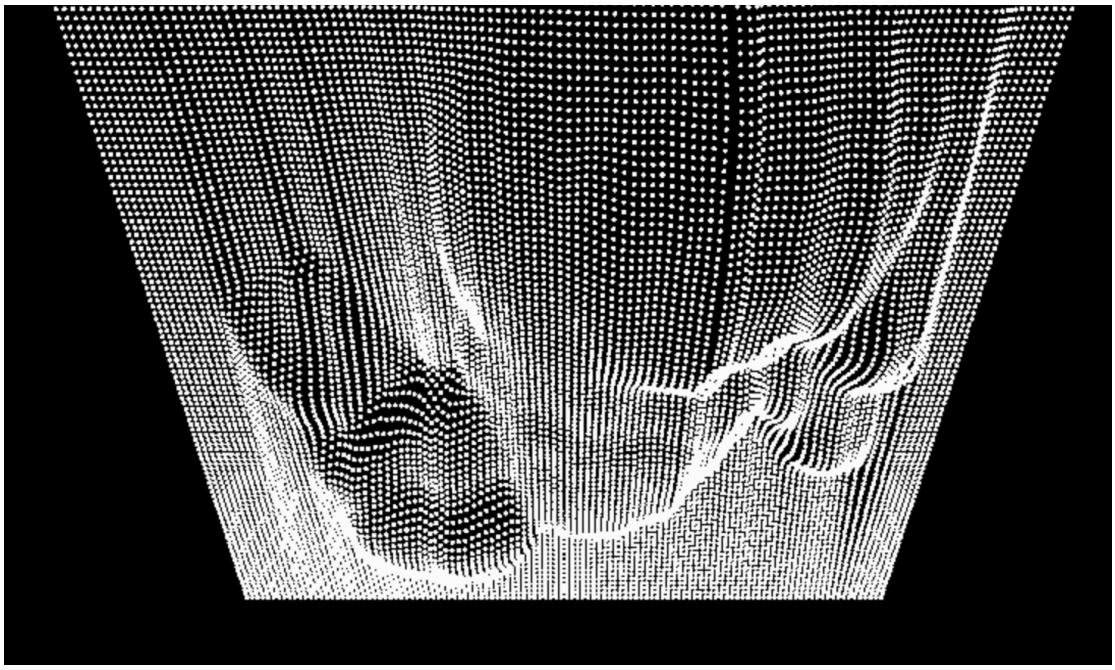
所以在使用方法三時，我將 **df/dy** 公式的負號拿掉，就正常了.

執行結果 ：

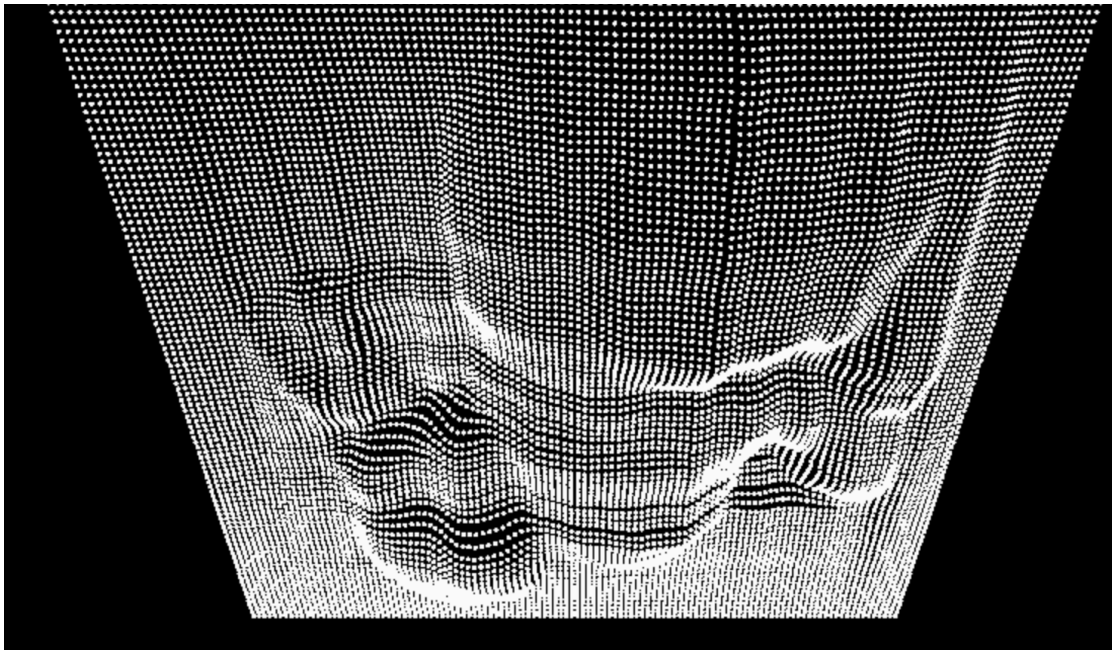
方法一（ 先積 **X** 再積 **Y**，上下方向混合 ） - **test/bunny**



方法二（ 先積 Y 再積 X, 左右方向混合 ） - test/bunny



方法三（ 整合方法一和方法二 ） - test/bunny



方法三（ 針對有雜訊的 **bunny** ）　　-　　**test/special/bunny**