

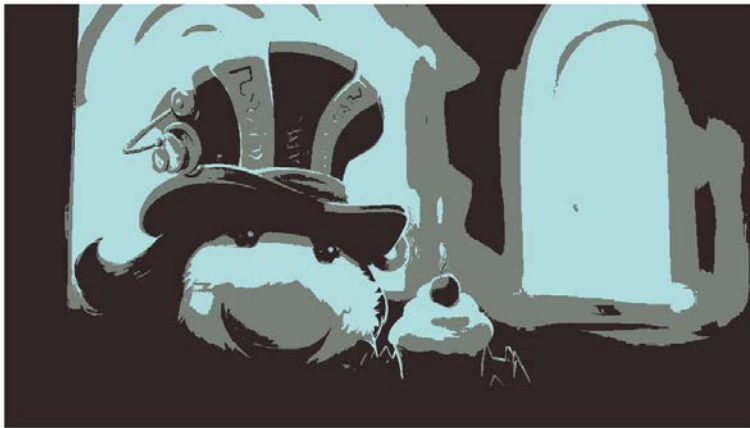
CV HW3 Report

蕭文逸 105062581

Part1

A: RGB with random initial guesses

(From top to bot: $K = 3, 7, 11$)



B: RGB with random manually initial guesses

(From top to bot: $K = 3, 7, 11$)



C:

LUV with random manually initial guesses

(From top to bot: $K = 3, 7, 11$)



LUV with random manually initial guesses



D: Discussion

一般而言，隨著 K 值上升，能表現的色彩越多樣。除此之外 $kmeans$ 的 $error$ 也會越小。至於手動或自動選點的結果差異不大，但是自動選點需要看隨機出來的起始點分佈夠不夠優良，不好的起始點或讓收斂次數或是 $error$ 增加，因此需要多次(50 random initial)的選取起始點才能有跟手動差不多的表現。

另外 LUV 在結果來看比 RGB 還要優良， $error$ 也大多比較小。

Part2

A: Background Replacement

我採用手動選擇起始點，並分成四類。(K = 4)

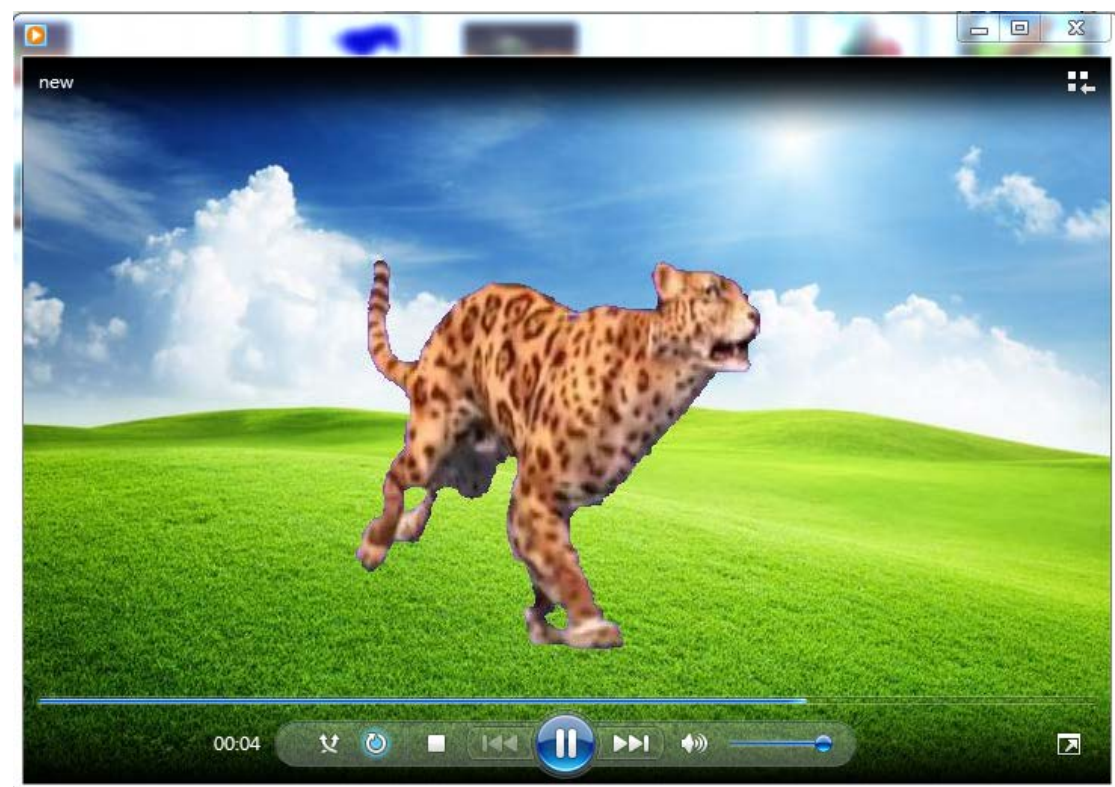
Label1:豹斑 (黑)

Label2:豹身 (棕)

Label3:背景 1 (深藍)

Label4:背景 2 (淺藍)

有了這四個起始就可以有很好的 $kmeans$ 結果，接著利用其結果對其他圖進行 clustering。並把其他圖中被分到 3 號 4 號的 pixel 用背景圖取代掉。

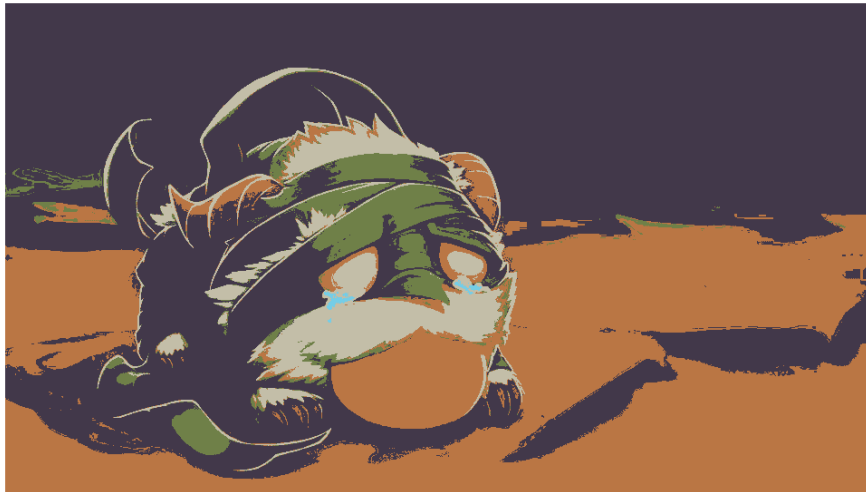
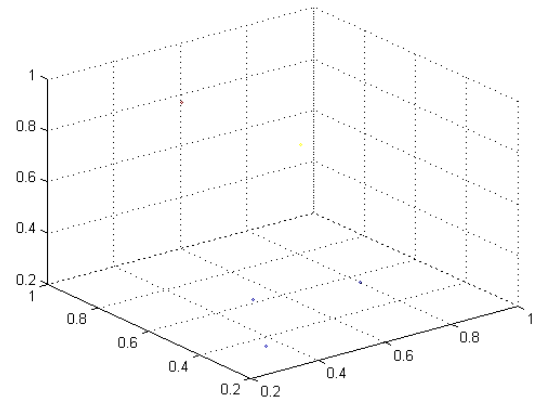
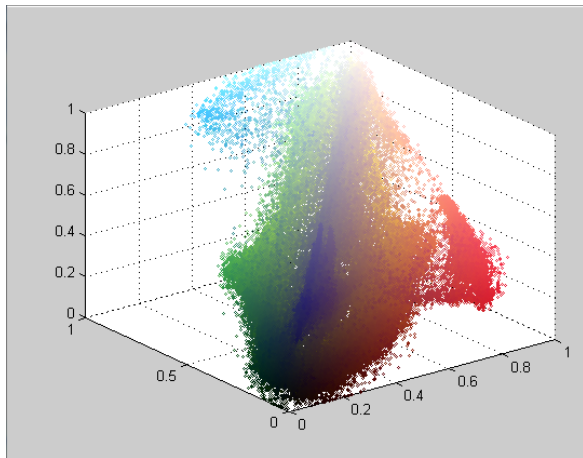


B: Accelerating

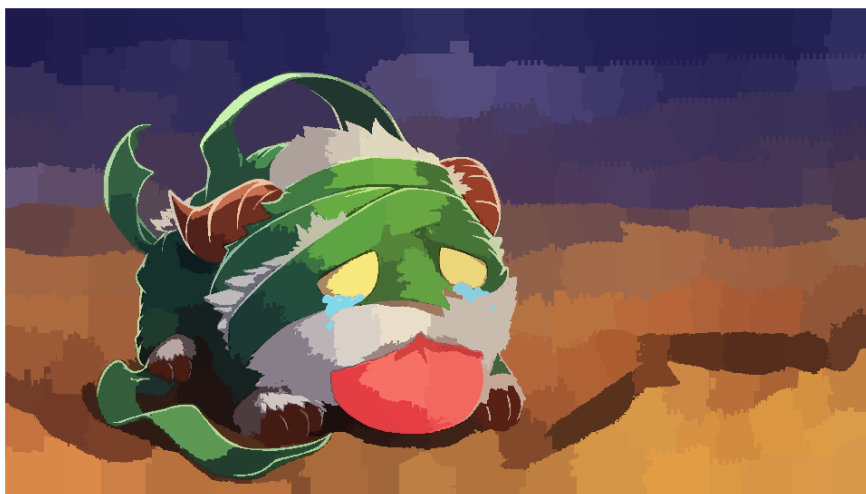
Matlab 對於迴圈運算很慢，因此我把運算矩陣化，把一張 $H \times W$ 大小的圖變成一個 $H*W$ 長的資料串，由兩層迴圈變成一層來加快速度。

Part3

A: RGB with min shift (bandwidth color = 0.19)

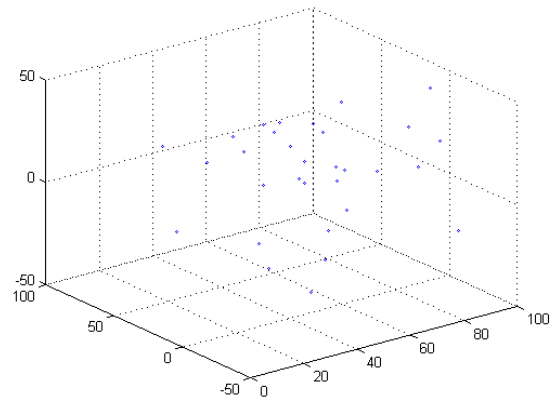
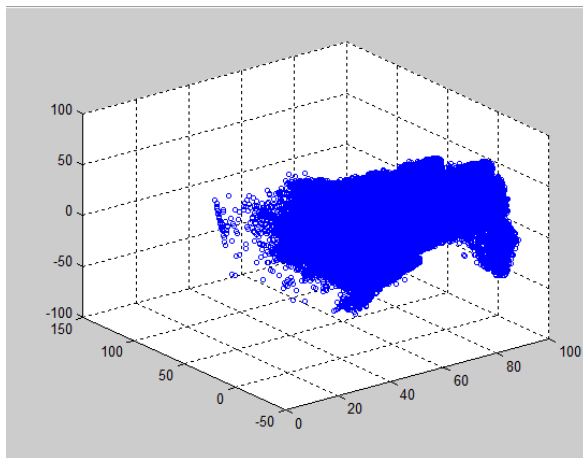


B: (bandwidth color = 0.19, bandwidth space = 50)



C: LUV with min shift

(A) (bandwidth color = 10)



(B) (bandwidth color = 10, bandwidth space = 20)

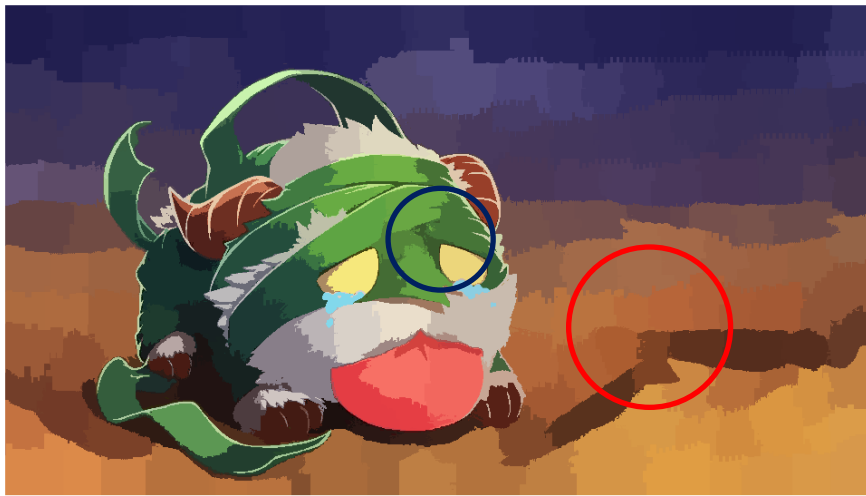


可以明顯看到兩者都有考慮到 **space** 的話，效果會比較好。像是下方的例子就很明顯，不考慮 **space** 只考慮色彩會出現舌頭的顏色出現棕色，而考慮到 **space** 後就不會有這種情形。

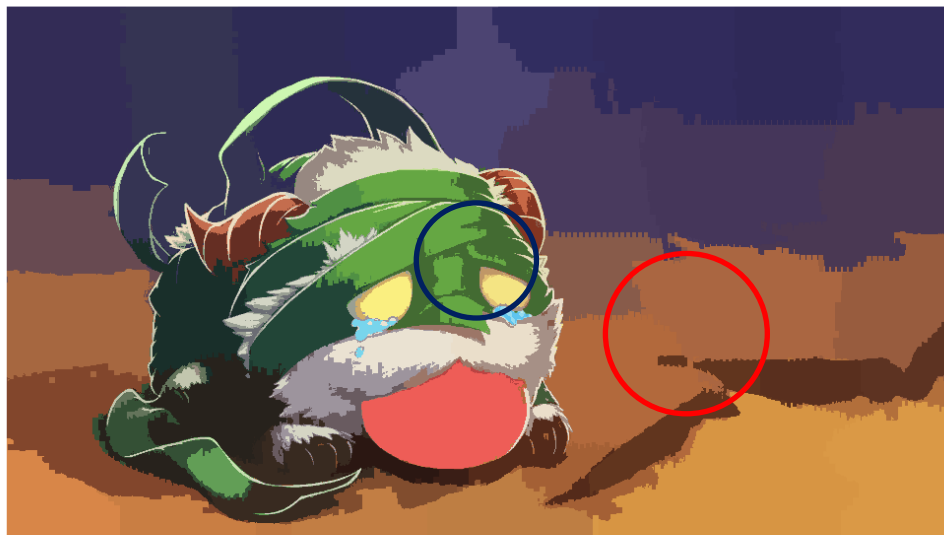
另外邊緣鋸齒是因為運算時我用方型去近似圓形範圍，並考慮相鄰和顏色相近點會有差不多的性質因此一起處理，我所以才會發生。

D: Comparison between RGB with different bandwidth

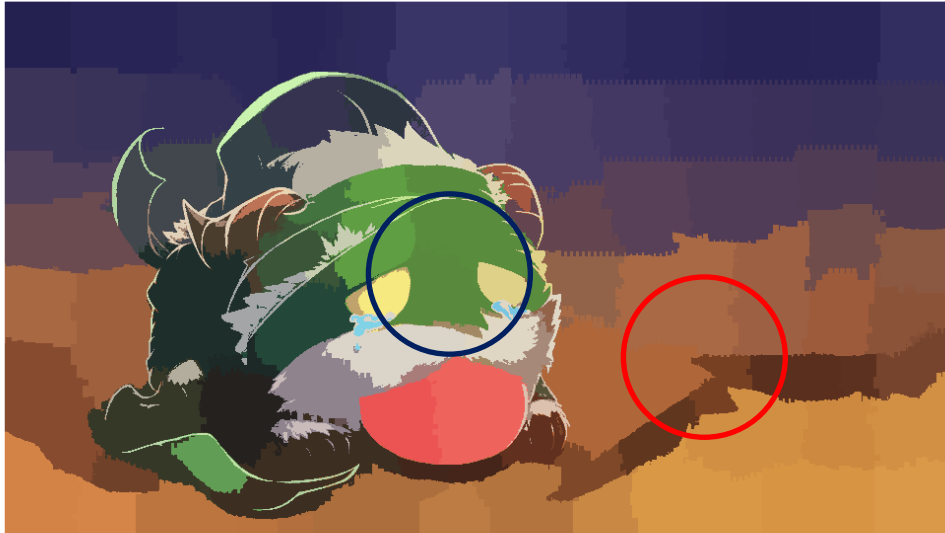
ms_rgb_50_space_50.png : (bandwidth color = 0.19, bandwidth space = 50)



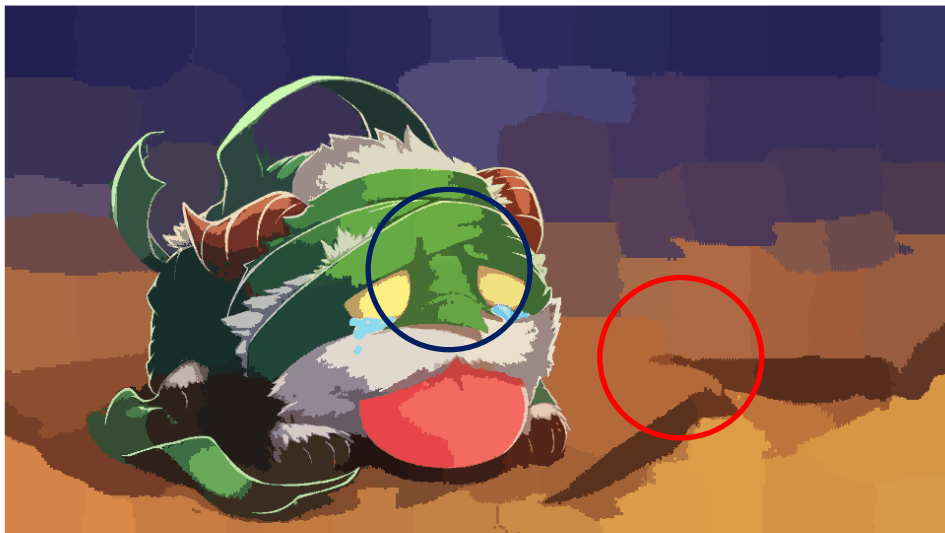
ms_rgb_50_space_30.png : (bandwidth color = 0.19, bandwidth space = 30)



ms_rgb_30_space_50.png : (bandwidth color = 0.11, bandwidth space = 50)



ms_rgb_30_space_30.png : (bandwidth color = 0.11, bandwidth space = 30)



隨著 space bandwidth 增加，空間細節就比較不會被保留，像是上圖紅圈的地方，可以看到當 bandwidth 小者，岩石交界處比較明顯
另外隨著 color bandwidth 增加，顏色細節就比較不會被保留，像是上圖黑圈的地方 bandwidth 小者，繃帶紋路或是眼影比較明顯。