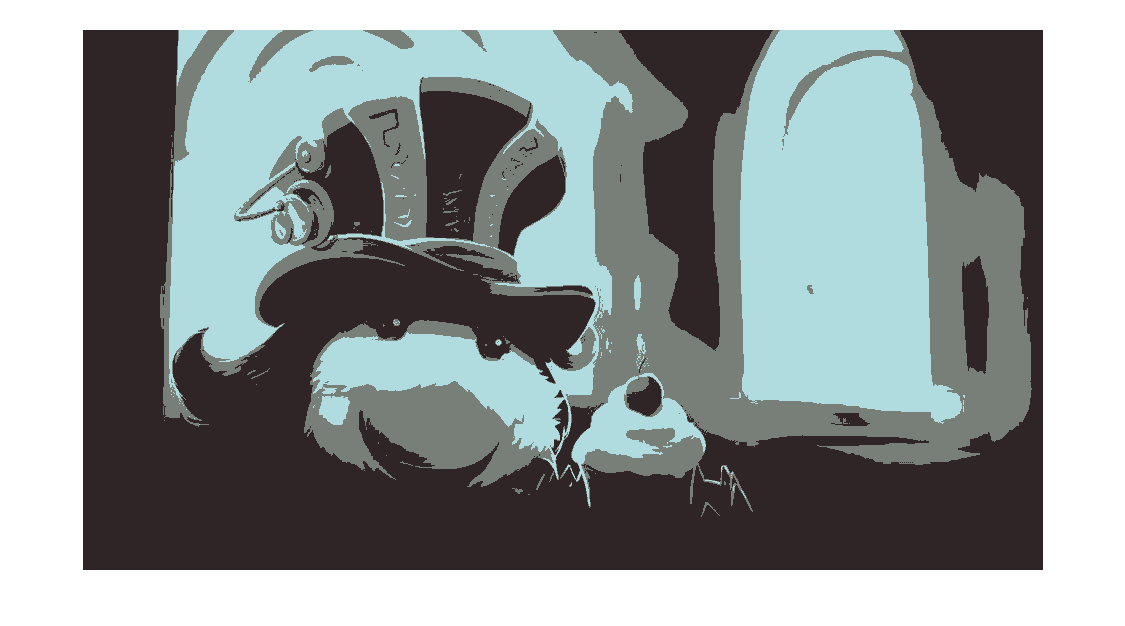
**CV HW3 Report 蕭文逸 105062581**

**Part1**

**A: RGB with random initial guesses**

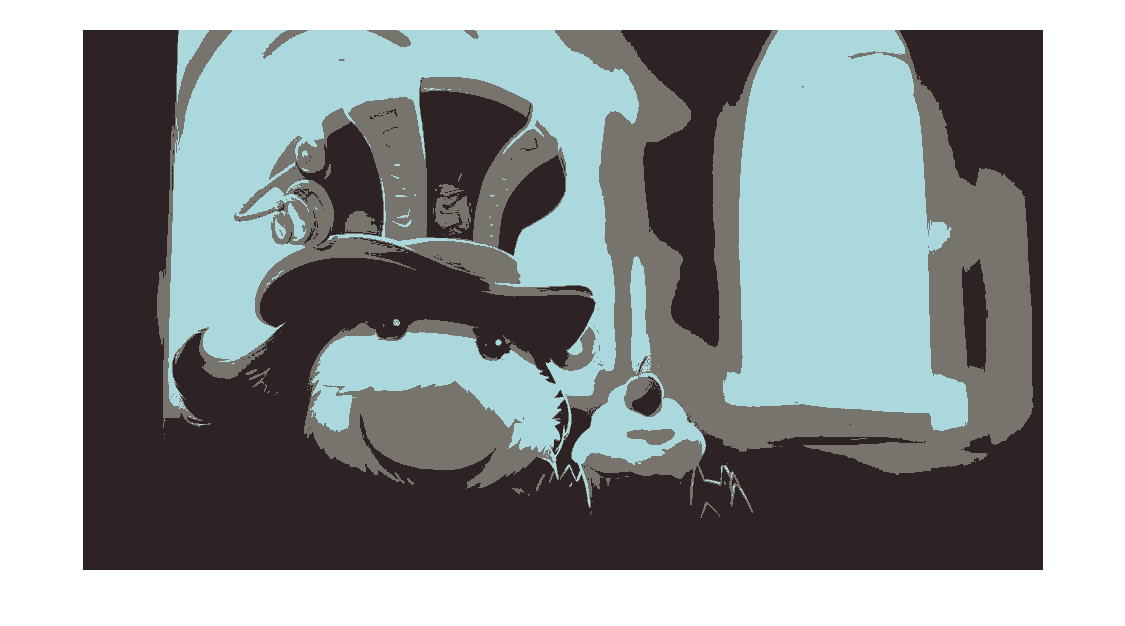
(From top to bot: K = 3, 7, 11)

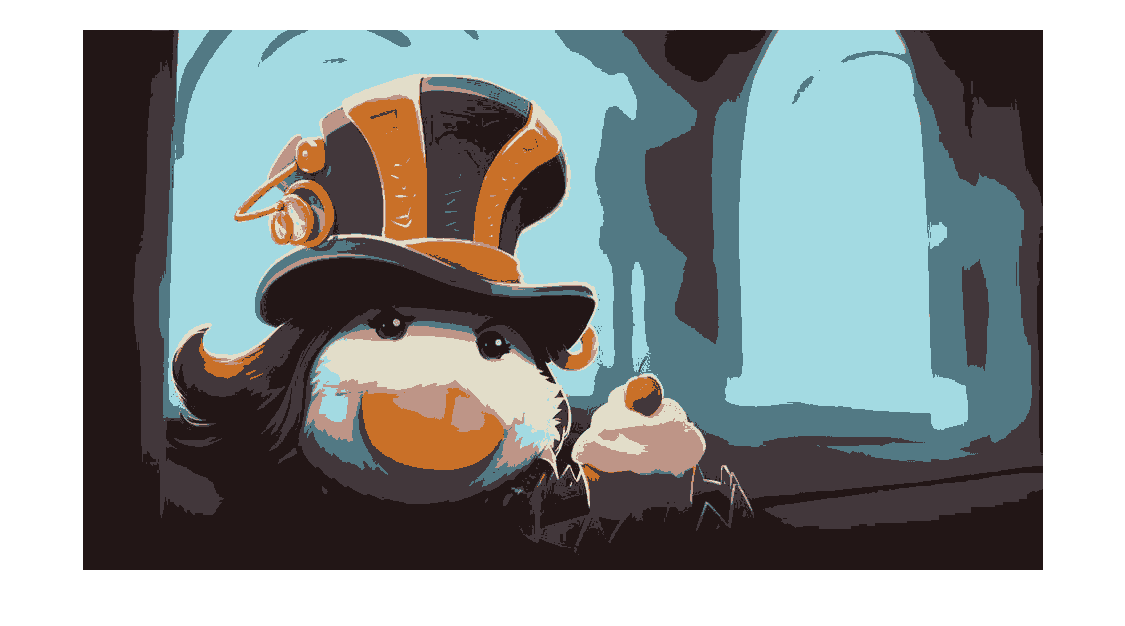




**B: RGB with random manually initial guesses**

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





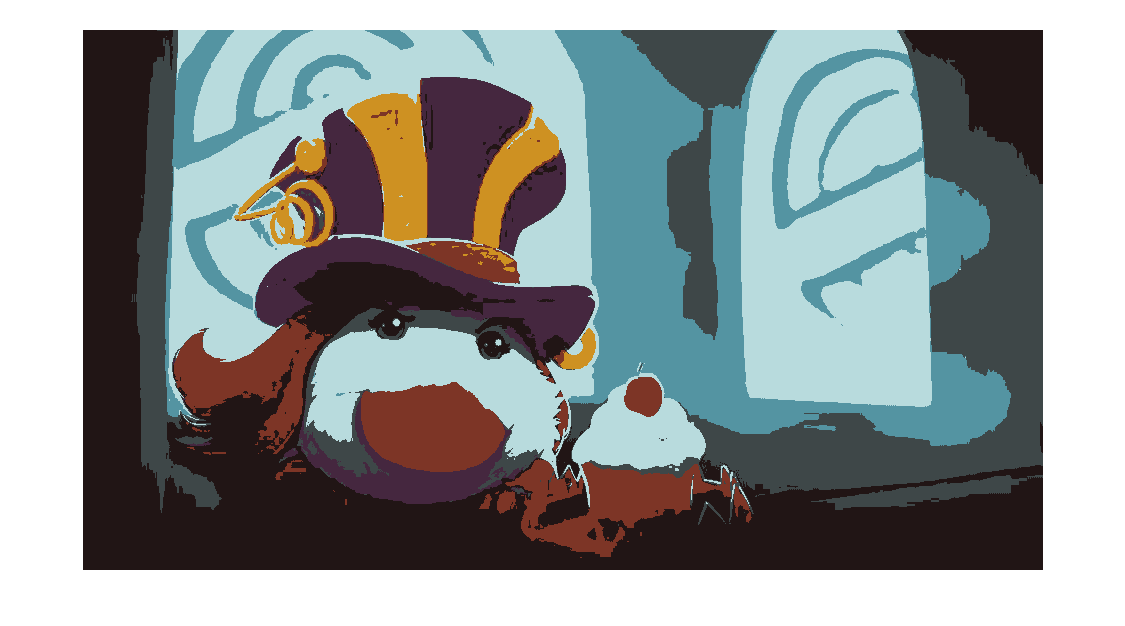


**C:**

**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**

我採取手動選擇起始點，並分成四類。(K = 4)

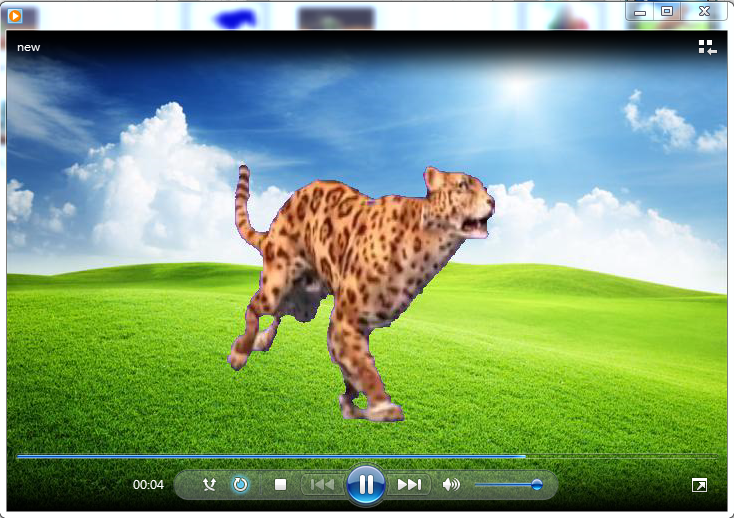
Label1:豹斑 (黑)

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。

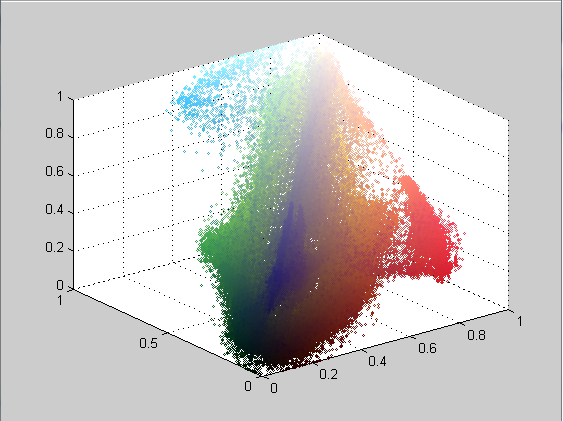
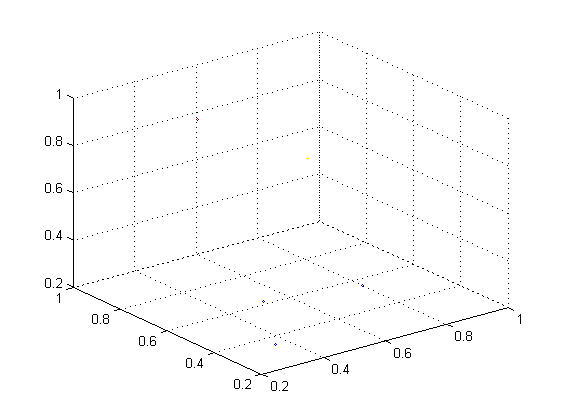


**B: Accelerating**

Matlab對於迴圈運算很慢，因此我把運算矩陣化，把一張H x 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)**

****

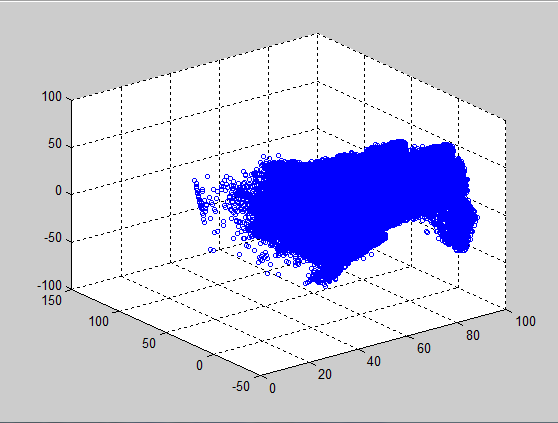
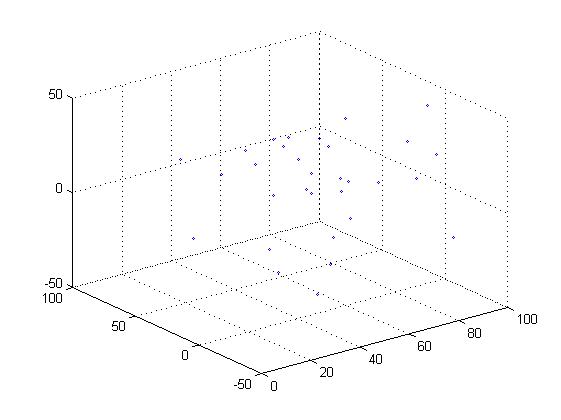
****

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

****

**C: LUV with min shift**

1. **(bandwidth color = 10)**

****

****

1. **(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，而老慮到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。