國立成功大學

資訊工程學系

多媒體內容分析

作業二

**基於高斯混合模型的彩色圖像分割**

***GMM-based Color Image Segmentation***

學生：P76094169 黃仁鴻

授課老師：朱威達 教授

中華民國110年4月

## 執行環境與說明

表 1環境配置

|  |  |
| --- | --- |
| **處理器** | Intel(R) Core(TM) i5-7400 CPU @ 3.00GHz |
| **記憶體** | 16.00GB |
| **作業系統** | Windows 10 64 位元作業系統 |
| **使用語言** | Rust 1.51.0 |
| **函式庫** | Image-rs、rusty-machine |

表1是本次作業的環境配置。使用rusty-machine中的gmm來進行此次作業的實驗，不過rusty-machine缺乏自訂初始參數的功能，因此將原版的rusty-machine fork下來做些微調整，使其可以載入預訓練的參數。

執行方法

* 訓練
  + hw2.exe train –h

查看訓練模式的使用說明

* + hw2.exe train –e <最大更新次數，可選，預設為10>  
     -k <kernel數目，可選，預設為2>  
     -w <儲存權重檔名，可選，   
     預設為gmm\_file>  
     -r [<場景分類參考用的輸入檔>  
     <場景分類參考用的標記檔>](至少一組)  
     -f <訓練用檔案，必要，可多個>
* 評估
  + hw2.exe eval –h

查看訓練模式的使用說明

* + hw2.exe eval –i <輸入檔案，必要>  
     -w <儲存權重檔名，必要>  
     -o <輸出分割圖檔名，可選，預設不儲存>  
     -t <標記圖檔名，可選，  
     有的話會計算 Dice Coefficient>

關於場景分類參考

雖然使用高斯混和模型可以依據色彩將圖片分割成多個區塊，但這些區塊並沒有特定的類別。在沒有標記解答參照的情況下是無法分辨哪些kernel被歸類在場景的，因此在訓練時會多使用一組(或多組)輸入、標記，藉由GMM將「參照用輸入圖檔」分割的結果來分類每個kernel是否屬於場景類別。

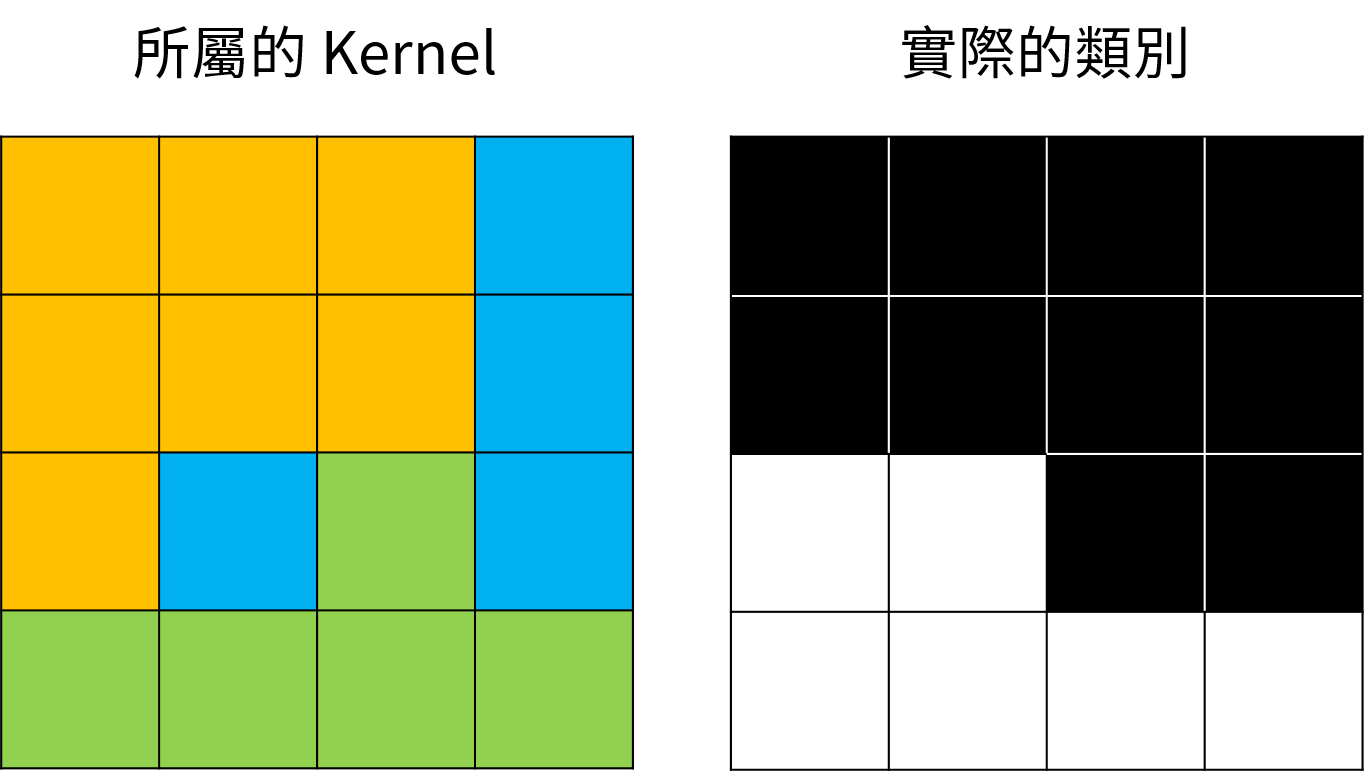


圖 1對參考圖片使用GMM分割

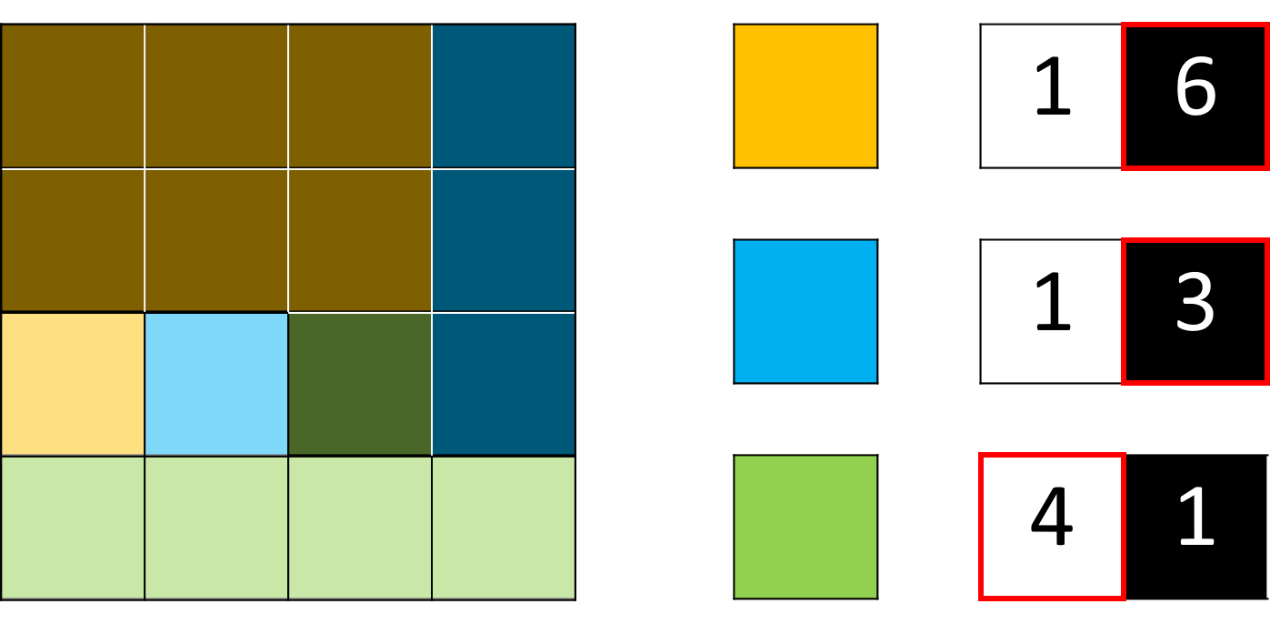


圖 2將分割結果與標記答案進行比對，依照每個kernel  
對應到的標記數量進行分類。

## 效能分析

由圖3可以發現，用較多資料訓練的model 2在不同kernel數量時都有相近的正確率，其表現相對穩定。而在只使用soccer 1訓練的model 1，在kernel數量增加時會有較明顯的性能提升。

圖 3比較使用於soccer 1時，model 1與model 2的效能差異。

因為soccer 1與soccer 2的明暗差異較大，用soccer 1訓練出來的模型難以兼容soccer 2，即使提增加kernel的數量，能提升的效果也有限(如圖4)。

圖 4比較使用於soccer 2時，model 1與model 2的效能差異

Model 2使用了兩筆資料進行訓練，然而因為soccer 2的場景因明暗度較為複雜，導致切割能力較低。但是當kernel增加時，就能讓GMM有更多能力擬合soccer 2使其效能有所提升(如圖5)。

圖 5比較model 2對soccer 1與soccer 2的效能差異

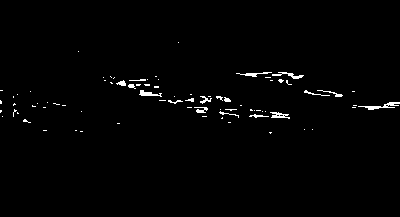






圖 6 model 1對soccer 2的切分結果，上圖是kernel=4，中圖是kernel=8，下圖是kernel=16。soccer 1與soccer 2的顏色分布差異較大，由soccer 1訓練的模型難以兼容至soccer 2。







圖 7 model 2對soccer 2的切分結果，上圖是kernel=4，中圖是kernel=8，下圖是kernel=16。可看出在較複雜的圖片上，增加kernel的數量有助於提升效能。