

第一部份：實驗方法與步驟

(1)使用 paint.net 圖像處理軟體與 windows 7 小畫家，將鴨子與非鴨子的圖像截取出來做為訓練資料，大部份鴨子的類別為 15x15px 的矩形圖像，而非鴨子的類別多為 100x100px 的矩形圖像。

(2)對每個像素點之 RGB 值儲存成三個維度的向量，使用迴圈進行掃描將所有鴨子的像素儲存成向量陣列，將 target 標記為 1，同理也將非鴨子的像素存成向量陣列，將 target 標記為 0，最後轉換成 NumPy 格式以利計算處理。

(3)將資料分成訓練資料以及測試資料，並且將資料順序打亂以利做訓練，接著把 data 與 target 輸入給 Gaussian Naive Bayes 分類器做訓練，最後與測試資料做比對計算正確率。

(4)將需要分類的圖像輸入給已訓練好的分類器做預測，依照要求將分類為非鴨子的像素標註成黑色 RGB(0,0,0)。

(5)將分類完成的圖像輸出，並計算總執行時間。

第二部份：實驗成果介紹

(1)實驗圖片輸出結果 (可從資料夾中開啟原始圖片 Images/Output)

如圖 1 所示，左圖為原始圖片，中圖為將分類為非鴨子的像素標為黑點之圖片，右圖為將分類為鴨子的像素標為紅點之圖片。

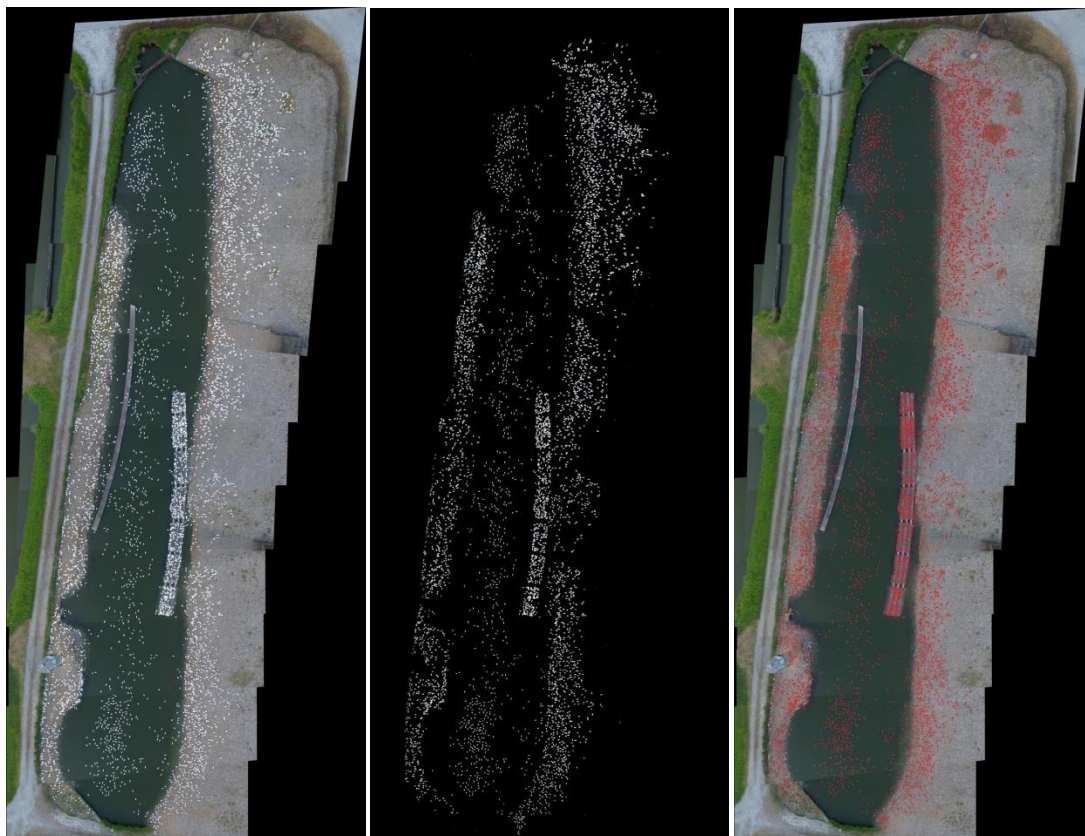


圖 1 原始圖片(左)、分類後之圖片(中、右)

(2)資料與成果分析

表1為訓練樣本的數量，收集鴨子的像素相對於非鴨子來說較為困難，實驗時鴨子樣本較少，這裡假設 $P(\omega_0) = P(\omega_1)$ ，因此對結果影響並不大。

表 2 為訓練資料與測試資料之比較，準確度一般能達到 99%。

表3、表4為實驗時的程式執行時間測試，輸入圖片大小為5946px * 13816px，測試環境如下：AMD Athlon X4 860K、8GB RAM、Windows 7 64bit。

	鴨子	非鴨子
樣本數量(px)	13095	251250

表 1 訓練樣本數量

	測試資料大小 30%	測試資料大小 50%	測試資料大小 70%
準確率	99.87%	99.87%	99.87%

表 2 分類器準確率

	第一次	第二次	第三次	第四次
執行時間	164 秒	164 秒	163 秒	169 秒

表 3 程式執行時間(將非鴨子標記為黑點)

	第一次	第二次	第三次	第四次
執行時間	86 秒	87 秒	88 秒	86 秒

表 4 程式執行時間(將鴨子標記為紅點)

第三部份：問題與討論

(1)Input 圖片大小為 5946px * 13816px，程式直接處理會導致執行過久或電腦崩潰。經過分析後猜測應為記憶體不足所導致，於是將程式經過修改，一次只儲存一定數量的資料，將其辨識後做即時處理，接著釋放記憶體後再進行下一行的處理。執行效能上經過測試一般在 3 分鐘內可以完成輸出。

(2)池塘邊石子堆的像素較容易被誤判為鴨子，在鴨子頭的像素也常被誤判為不是鴨子，在初期測試時訓練資料時硬性規定圖片大小為 15px * 15px，將程式修改後能達到輸入圖片不限大小、名稱、類型、也不需要設定張數，能方便使用者做新增及刪除的動作，在新增非鴨子的圖片時提高了不少便利性，也成功將雜訊變得更少。

(3)這次實驗將程式模組化就花了不少時間，其中將 function 以及 config 獨立出來的時間甚至比程式撰寫的時間還久，即使最後程式的執行結果並無區別，但程式碼能較方便的被調整、修改及使用。

第四部份：結論與心得

(1) 根據參考資料”應用圖形辨識於青萍數量之計算”[1]的做法，可收集訓練樣本後以人工計算的方式取得鴨子的平均像素數量，可用此值來估計預測目標的鴨子數量。而收集到的訓練樣本越多平均值也會越準確。

(2)若以此方法來預測鴨子數量，誤判石頭為鴨子的成本會較高。

(3)使用 `github` 在版本控制上很方便，也方便整理作品集。

參考資料：

[1]蕭友晉(Shiau, Yo-jin) 張文亮(Chang, Wen-lian) (20070300)。[應用圖形辨識於青萍數量之計算]。《數位典藏與數位學習聯合目錄》。

<http://catalog.digitalarchives.tw/item/00/51/a5/90.html> (2018/11/29 瀏覽)。