多媒體技術概論 – 作業一

資工三 404220029 陳麥克

* 方法描述 - 演算法原理與實作方式描述

1. sum of absolute distance



將圖像中每個像素的值相減後取絕對值再相加，這樣可以找出差距自己最少的圖片。

2. sum of square distance

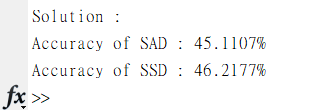


將圖像中每個像素的值相減後平方再相加，這樣可以找出平方後差距自己最少的圖片。

* 執行方式 - 執行的函數名稱、參數設定等

1. dir() : 這個指令會回傳一個結構，內容包含了所有檔案的名稱、修改日期、大小等等，可以利用這個指令讀取路徑下的所有檔案。
2. strcmp() : 輸入兩個字串，相同的話return 1，否則return 0。
3. disp() : 類似於C# 中的printf。
4. zeros() : 產生一個陣列，元素都是0。

* 實驗結果 - 每一個階段的圖片、數據結果



* 結果討論 - 對於實驗結果的一些解釋和討論

原本想說用SSD、SAD可以做出80%以上的準確率，但是卻只有不到一半的準確率，但想想也對，畢竟如果用這麼簡單的方法就做出極高的準確率，那還需要人工智慧的人臉辨識嗎?

* 問題討論 - 作業撰寫中遭遇的演算法問題與實作的困難

問題: 不知道要怎麼讀取資料夾下的所有圖檔。

解決方法: 查詢資料後，發現可以用dir(路徑)這個指令，這個指令會回

傳一個結構，內容包含了所有檔案的名稱、修改日期、大小

等等，可以利用這個指令讀取所有圖檔。

問題: 讀進來的圖檔並沒有RGB 3個值，只有一個值。

解決方法: 圖檔似乎原本就已經是灰階的，wiki上寫說pgm格式是可移

植灰度圖格式，所以我沒有再把圖案做任何的轉換，直接做 SSD、SAD。

問題: SAD準確率 = 5.6273%。

解決方法: 發現原因是出在圖檔讀進來的資料型態是unsigned int，在做

SSD或SAD如果相減後是負值會變成0，這樣子會使得準確

率大幅降低，所以我把型態轉換成int8，就可以算負值了。

問題: SSD準確率太低。C:\Users\麥克\AppData\Local\Microsoft\Windows\INetCache\Content.Word\1.png

解決方法: 似乎是因為在做SSD時int8裝不下，所以overflow了，我把

型態轉換成int16後，準確率提高至40多%。

* 程式執行方法

必須要把CroppedYale資料夾跟作業檔案放在一起，才可以正常執行，如下



* 心得

這次的作業讓我複習了以前學過的matlab，雖然因為很久沒用matlab，出現了很多錯誤，但是也因此看到更多的風景，學習到更多東西，而對matlab更加的熟悉。