611021208 邱創業

1. 如何完成作業的報告
2. 依據資料庫的影像格式，設計一個讀取pgm影像檔的函式。

* 利用Pillow.Image 讀入.pgm影像檔，並show出相片。

一張含有 文字, 室內, 個人, 擺姿勢 的圖片

自動產生的描述

1. 每張人臉影像均為92x112=10304的灰階影像，讀取後請將其轉為10304x1的向量，即成為一個樣本。

* 讀入影像檔後，將其轉為numpy array並flatten。

一張含有 文字 的圖片

自動產生的描述

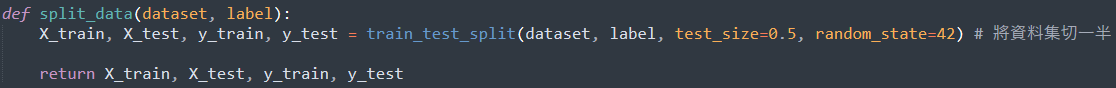
1. 資料庫共含有400張影像（40人，每人10張），訓練時請只用200張（每人取5張）。

* 利用os.listdir找出資料夾內所有檔案，並製作dataset以及label。

一張含有 文字 的圖片

自動產生的描述

* 使用train\_test\_split，將200張影像檔用於訓練，200張影像檔用於測試。



1. 利用PCA計算此200張影像的轉換矩陣，設法將維度從10304降至10, 20, 30, 40, 50維

* 使用sklearn. Decomposition.PCA將影像檔降維

一張含有 文字 的圖片

自動產生的描述

1. 以這些較低維度的樣本訓練出你所學過的任何分類器來進行辨識。

* 使用SVC訓練來進行辨識

一張含有 文字 的圖片

自動產生的描述

1. 請比較不同維度的辨識率，並統計出混淆矩陣(Google一下這是什麼？)

* 使用matplotlib來畫出混淆矩陣

一張含有 文字 的圖片

自動產生的描述

1. 請以降維後的樣本，繼續利用FLD(LDA)找出另一轉換矩陣，利用此矩陣轉換降維後的樣本（毋需降維只須轉換）為有較佳的類別分離度之新樣本。

* 使用sklearn.discriminant\_analysis. LinearDiscriminantAnalysis來進行LDA

一張含有 文字 的圖片

自動產生的描述

1. 結果

* PCA和LDA各個維度的混淆矩陣

|  |  |  |
| --- | --- | --- |
|  | PCA | LDA |
| 10 |  |  |
| 20 |  |  |
| 30 |  |  |
| 40 |  |  |
| 50 |  |  |

* 各個維度的識別率

|  |  |  |
| --- | --- | --- |
|  | PCA | LDA |
| 10 | 85.00% | 93.00% |
| 20 | 88.50% | 96.50% |
| 30 | 91.00% | 96.00% |
| 40 | 90.00% | 94.00% |
| 50 | 90.00% | 96.00% |

1. 討論結果

在實作此次作業的過程中發現降維的不同，會影響結果的表現。單純比較PCA及LDA，會發現LDA整體表現都較為優異，結果為LDA的20維表現最好。

1. 總結

在此次作業當中學習到了許多新的程式技巧，如何讀取資料夾內的所有影像檔、切割資料、降維、建立模型以及畫出混淆矩陣。我個人覺得老師的課程作業相對地比較困難，但是學到的東西也比較多，需要多花時間與功夫下去做。