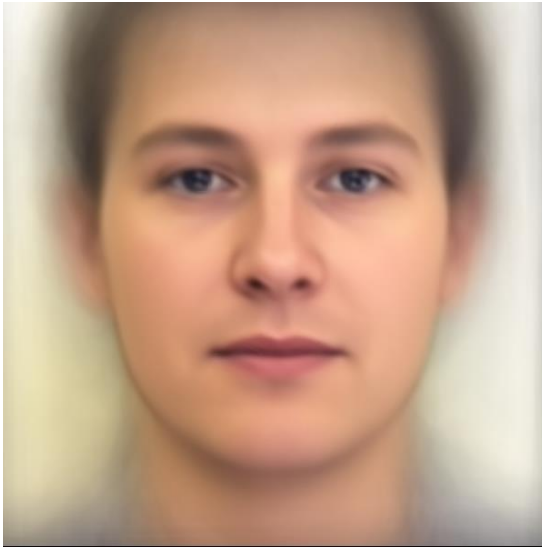


## Machine Learning HW7 Report

學號：B04901069 系級：電機四 姓名：林志皓

### 1. PCA of color faces:

- a. 請畫出所有臉的平均。

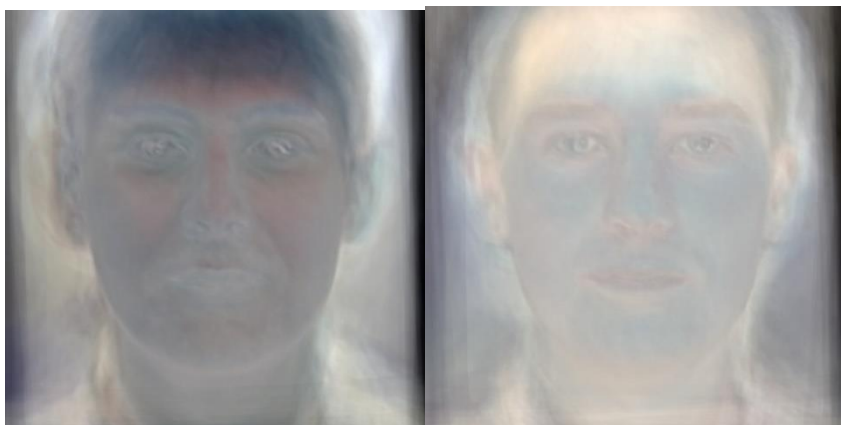


- b. 請畫出前五個 Eigenfaces，也就是對應到前五大 Eigenvalues 的 Eigenvectors。



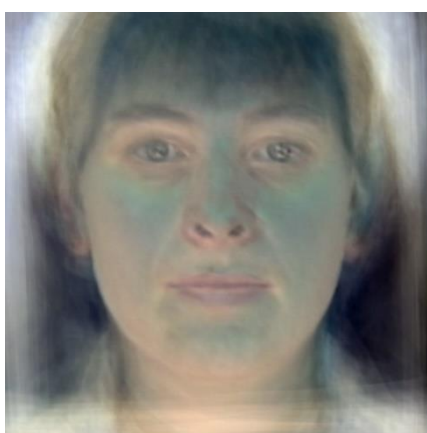
eigenface 1

eigenface 2



eigenface 3

eigenface 4



eigenface 5

- c. 請從數據集中挑出任意五張圖片，並用前五大 Eigenfaces 進行 reconstruction，並畫出結果。

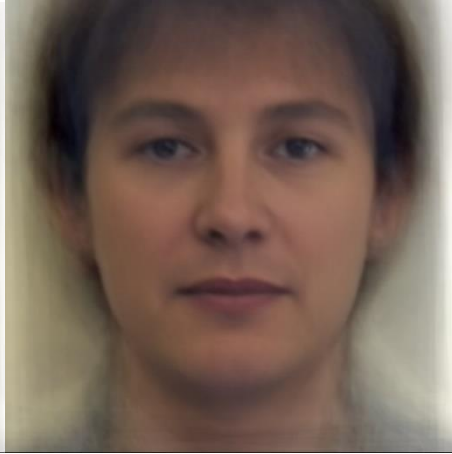


original

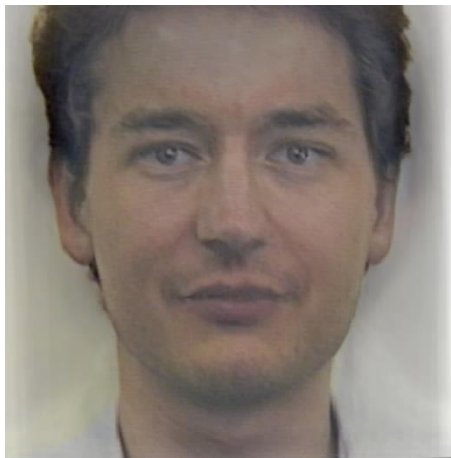
reconstruction



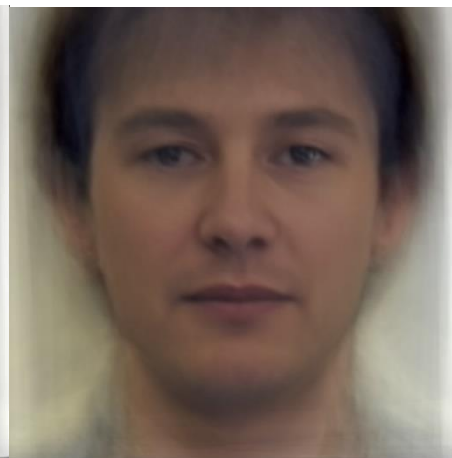
original



reconstruction



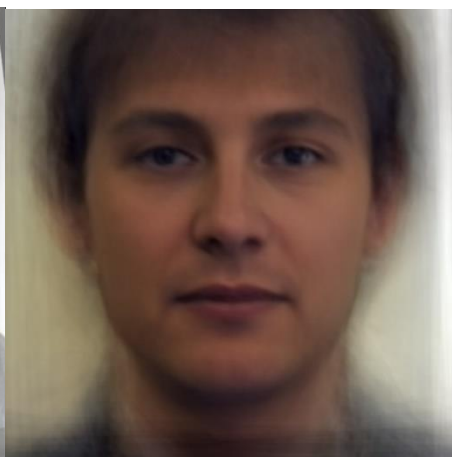
original



reconstruction



original



reconstruction



original

reconstruction

d. 請寫出前五大 **Eigenfaces** 各自所佔的比重，請用百分比表示並四捨五入到小數點後一位。

(1) 4.1%, (2) 2.9%, (3) 2.4%, (4) 2.2%, (5) 2.1%

## 2. Image clustering:

a.請實作兩種不同的方法，並比較其結果(reconstruction loss, accuracy)。(不同的降維方法或不同的 **cluster** 方法都可以算是不同的方法)

作法一：

我利用 **autoencoder** 進行訓練，訓練好之後將圖片降維成 512 維，直接進行 **k-means clustering**, 分成 8 群，我利用自己標的前一百張作為參考，挑出為一兩個 **dataset** 所屬的群，然後依此對剩下的進行分群，得出的 **accuracy** 大約為 0.88

作法二：

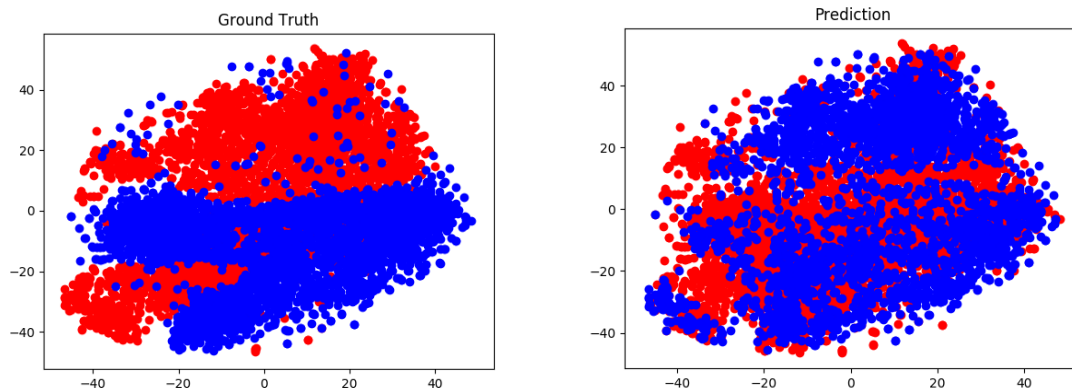
利用與作法一 一樣的 **autoencoder** 將圖片降為 512 維，然後利用 **sklearn** 的 **PCA** 套件，將 512 維更進一步壓縮為 128 維，最關鍵的部份是設定參數'**whiten = True**'，再進行 **k-means clustering**, 分為兩群。如此一來 **accuracy** 為 0.955

b. 預測 **visualization.npy** 中的 **label**，在二維平面上視覺化 **label** 的分佈。

(用 **PCA**, **t-SNE** 等工具把你抽出來的 **feature** 投影到二維，或簡單的取前兩維 2 的

feature)

其中 `visualization.npy` 中前 2500 個 images 來自 dataset A，後 2500 個 images 來自 dataset B，比較和自己預測的 label 之間有何不同。



我將所有圖片(包含 `visualization.npy`)利用 `encoder` 壓縮到 512 維，再一起用 `pca` 壓縮到 128 維後進行 `k-means clustering` 分成兩堆，最後利用 `t-sne` 將 `visualization.npy` 的部份壓縮成 2 維。左圖為 `ground truth`，右圖為我的 `prediction`。兩種顏色分別代表兩種 `dataset`。可以看出我的模型預測大致分的正確，但在交界處因為距離兩種 `dataset` 的中心相近，因此被分錯。

- c. 請介紹你的 `model` 架構(`encoder`, `decoder`, `loss function`...)，並選出任意 32 張圖片，比較原圖片以及用 `decoder reconstruct` 的結果。

**Encoder 架構：**

全為 `convolutional layer`, `kernel size`= 4, `stride`= 2, `padding`= 1, 如此一來每經過一層長寬就會縮小一倍，`channel number` 分別為 3, 32, 64, 128, 256, 512。最後 `output` 是 512 維的向量，由這個向量進行 `clustering`。

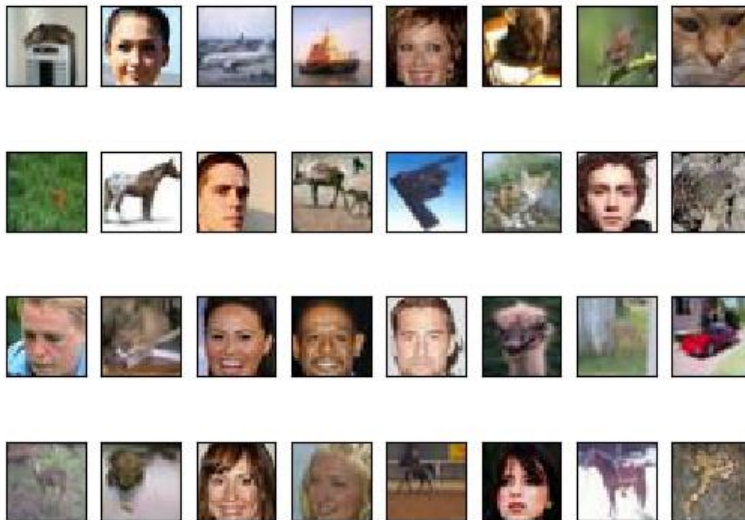
**Decoder 架構：**

全為 `deconvolutional layer`, `kernel size` = 4, `stride` = 2, `padding` = 1,如此一來每經過一層長寬就會放大一倍，`channel number` 分別為 512, 256, 128, 64, 32, 3。最後 `output` 是 32\*32\*3 的圖片。



loss function 為 MSE (Mean square error)

我所挑選的 32 張原圖：



模型重建的圖：

