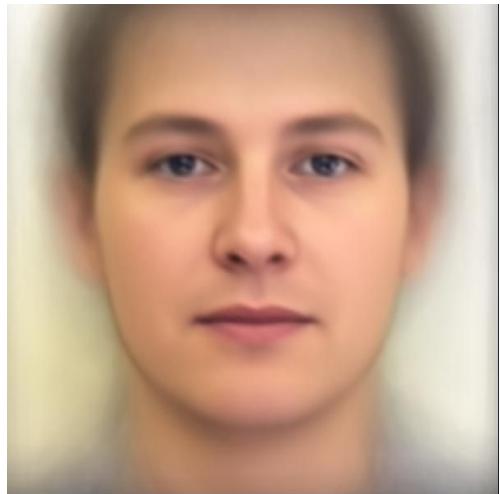


Machine Learning HW7 Report

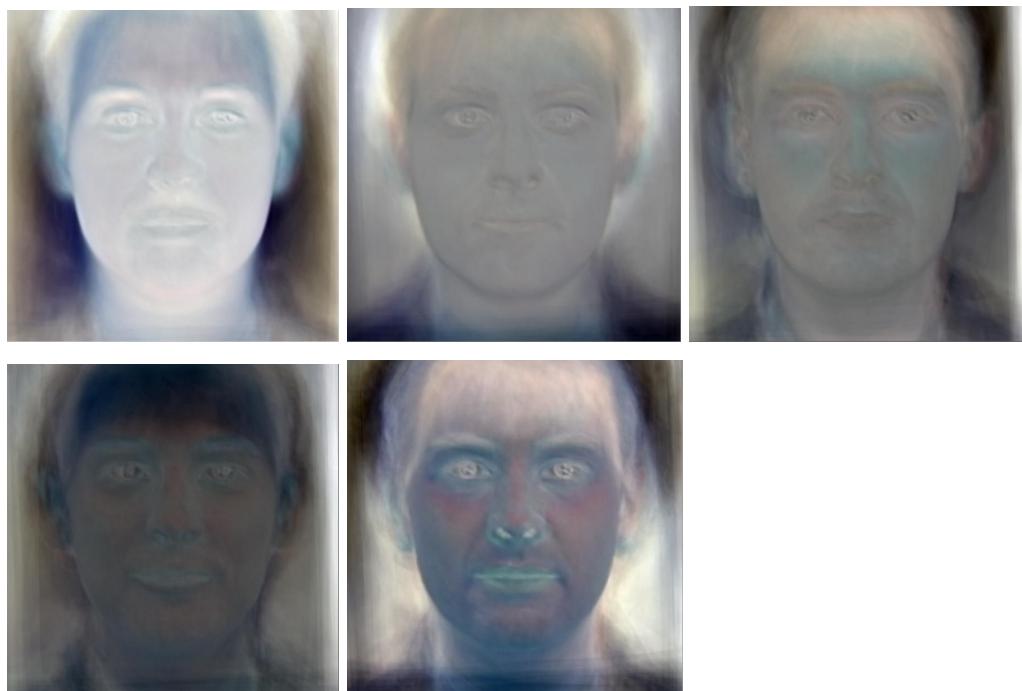
學號：R07922004 系級：資工碩一 姓名：吳星耀

1. PCA of color faces:

- 請畫出所有臉的平均。

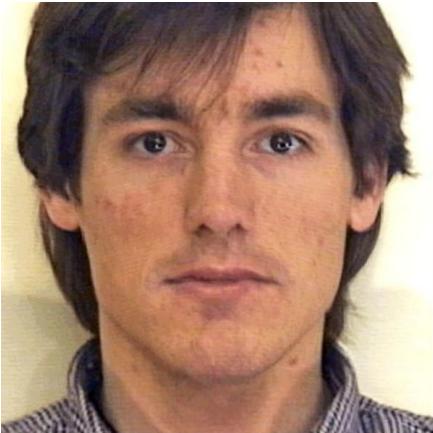
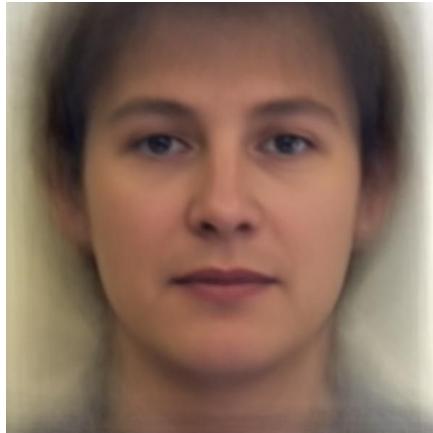
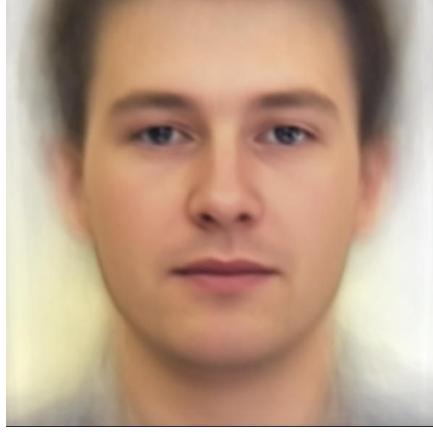


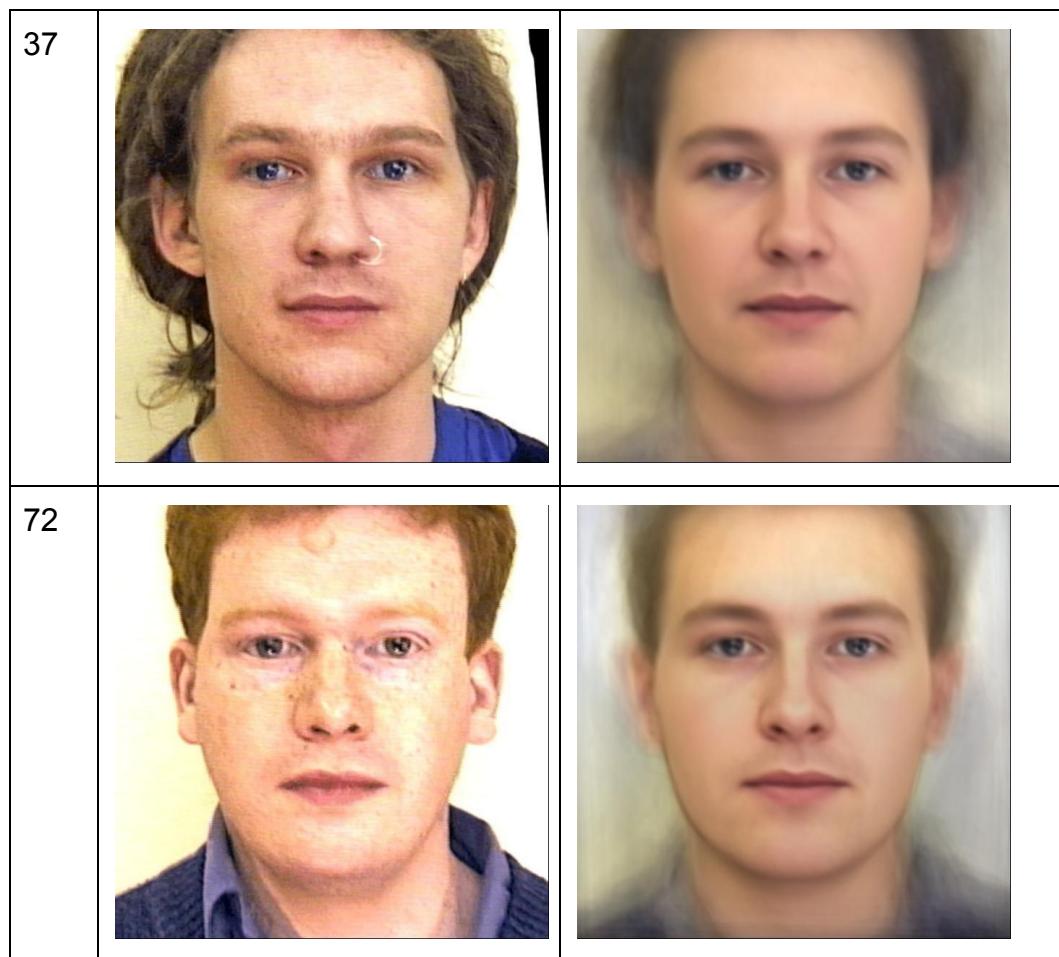
- 請畫出前五個 Eigenfaces，也就是對應到前五大 Eigenvalues 的 Eigenvectors。



由左而右，由上到下分別對應eigenvalue最大、次大、第三大、第四大、第五大的eigenface。

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。

No.	Original Image	Reconstructed Image
1		
10		
22		



- 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。

Eigenface					
比重(%)	4.1%	2.9%	2.4%	2.2%	2.1%

2. Image clustering:

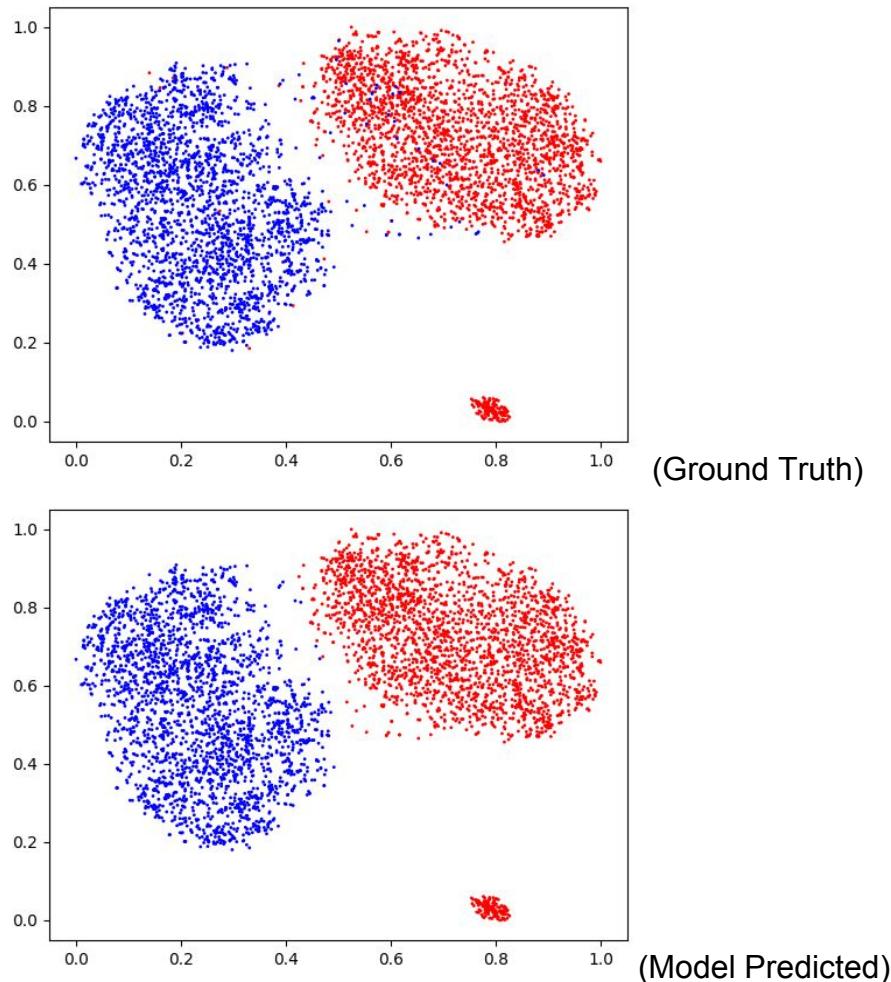
- a. 請實作兩種不同的方法，並比較其結果(reconstruction loss, accuracy)。
(不同的降維方法或不同的 cluster 方法都可以算是不同的方法)

我使用兩種不同latent dimension的autoencoder (分別是32維與128維)搭配t-SNE降到2維後以K-menas分群。其中兩種autoencoder都是使用4層Convolution layer和一層fully connected layer, 兩者唯一差別只在最後fully connected layer的output維度。

method	32 dim autoencoder	128 dim autoencoder
reconstruction loss	0.040801	0.019355
accuracy (public score)	0.97462	0.94181
accuracy (private score)	0.97433	0.94136

由上面比較可以發現，雖然32維的autoencoder因為能儲存的訊息比128維少，在reconstruction loss上不如128維，但是或許也因此它能篩選的出比128更重要的訊息，導致accuracy比128維還高。

- 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。



紅色代表來自dataset A，藍色代表來自dataset B。由圖可以發現明顯分成兩群，但是其中有少數紅色點被分到藍色群，少數藍色點被分到紅色群中。這些可能是因為太過於不像人臉的人臉或者太像人臉的其他圖所導致。另外在右下角可以發現有一小團紅點聚集，猜測這些點可能是側臉的圖片，因為與正臉圖片有些差距，因此無法與正臉圖片歸類到同一個群之中；也有可能是來自於黑人人臉與其他種族的人臉膚色差距過大，導致這種分群，或者也有可能是其他的差距導致。

- 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。

我的encoder使用4層Convolution layer和一層fully connected layer，每個Convolution layer由一個Conv2d、InstanceNorm2d與LeakyReLU(2e-3)組成，每層channel與filter數量分別為 (8, 3), (16, 4), (32, 4), (64, 4), stride為2，並且層與層之間不加Maxpooling。fully connected layer的input數量為

$64 \times 4 \times 4$, output 數量為 32, 後接一層 LeakyReLU(2e-3)。decoder 則是與 encoder 幾乎相反, 由一層 fully connected layer 跟 4 層 Convolution layer 構成, 其中與 encoder 不同的地方為將 InstanceNorm2d 改為 Batchnorm2d, 將 LeakyReLU 改為 ReLU, 而 convolution layer 則使用 Pytorch 的 ConvTranspose2d。我使用 MSELoss() 作為 training 的 loss function。在通過 autoencoder 後, 我會再用 t-SNE 再從 32 維降到 2 維, 再做 K-means 分群。

以下為測試圖片與 autoencoder 的對應重建圖片：

原圖片：



Reconstruction：



由上面兩張圖可以發現：重建的圖都會變得模糊，除了人臉以外的圖片幾乎都很難被重建，有些不是圖重建後還會變得有一點點像人臉。人臉的圖片被重建後雖然大部份能夠看得出是人臉，但是細節也會出現變化：不笑的人重建後表情會變得像在笑，除了臉部以外的地方如頭髮、背景都會變得非常模糊，有些人臉的角度也會產生變化。總之可以判斷這個 Autoencoder 對於人臉的重建能力較強，對於其他圖片的重建能力較弱。