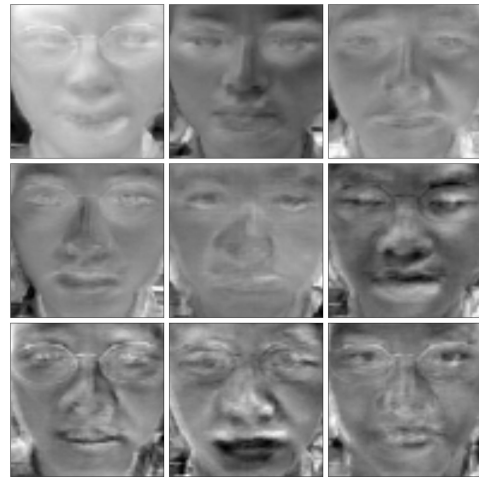
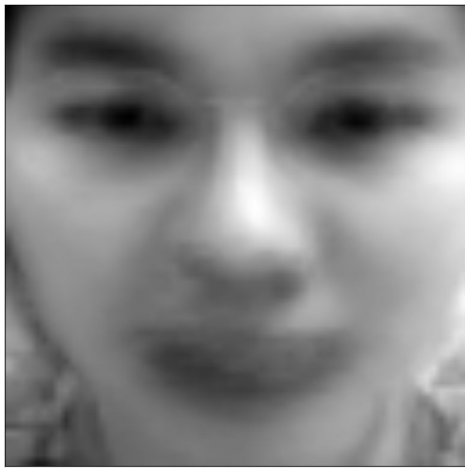


1.1. Dataset 中前 10 個人的前 10 張照片的平均臉和 PCA 得到的前 9 個 eigenfaces:

答：(左圖平均臉，右圖為 3x3 格狀 eigenfaces, 順序為 左到右再上到下)



1.2. Dataset 中前 10 個人的前 10 張照片的原始圖片和 reconstruct 圖 (用前 5 個 eigenfaces):

答：(左右各為 10x10 格狀的圖, 順序一樣是左到右再上到下)



1.3. Dataset 中前 10 個人的前 10 張照片投影到 top k eigenfaces 時就可以達到  $< 1\%$  的 reconstruction error.

答：k=59 (divided by 256); k=60 (divided by 255)

## 2.1. 使用 word2vec toolkit 的各個參數的值與其意義：

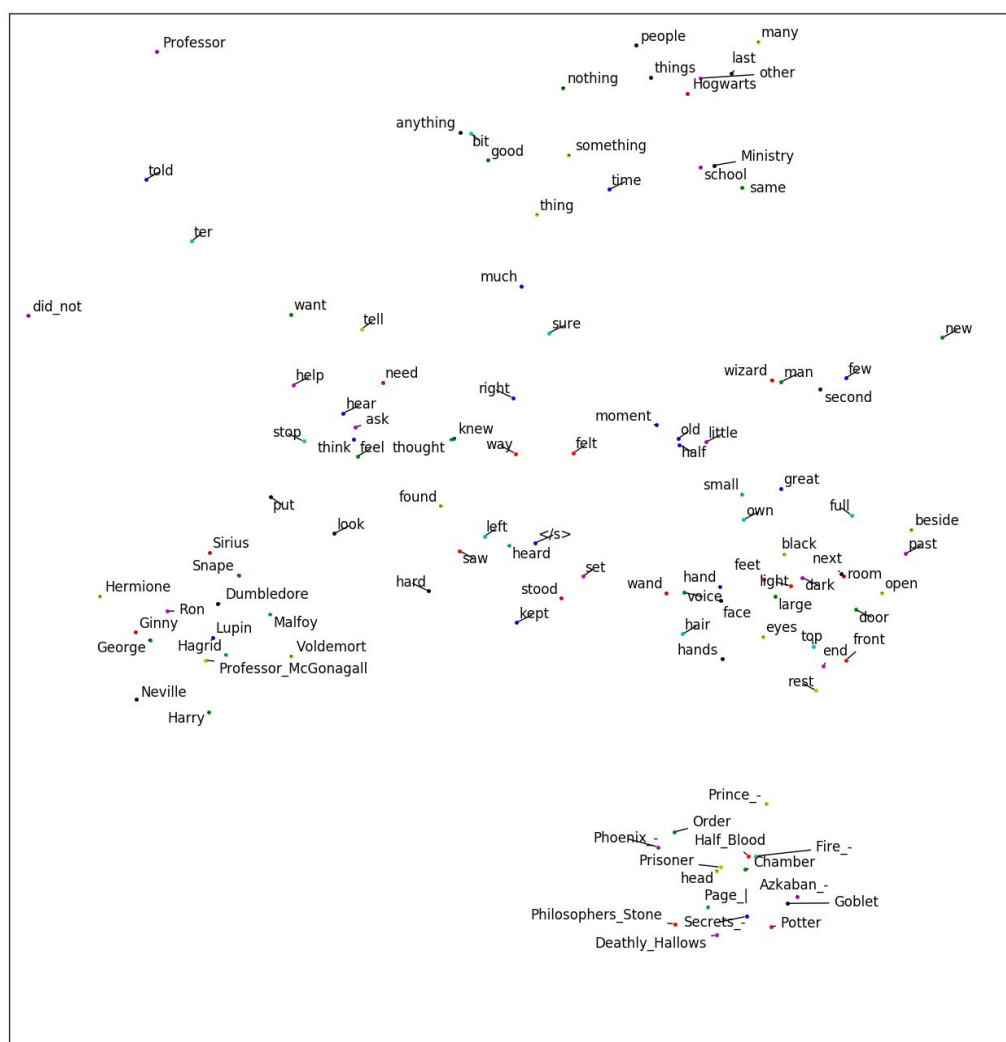
答：

word2vec 的參數中我只使用 train 和 output，其餘使用預設參數，但裡面還有很多其他參數，說明如後。size 就是轉成 vector 時的維度；window 是做 skip-gram 時往左往右看的最大長度，要設計成「最大」長度是因為遇到句尾或句首時，左右並沒有那麼多字，預設是 5；sample 是設定高頻字的門檻，超過此門檻的高頻字有機會被忽略，會需要這麼做的原因是，在訓練時若沒有時而略過不訓練這些高頻字，他們的 word vector 在空間中會亂飄，失去其意義，預設是 1e-3；hs 在 word2vec 中用 binary Huffman tree 實作，使得計算 softmax 可以更有效率，讓複雜度從  $O(V)$  變成  $O(\log(V))$ 。所以這個參數是決定要不要使用 hierarchical softmax，預設不使用；min-count 是設定一個門檻，出現次數小於此數字的字會被丟棄，預設是 5；alpha 是 learning rate 初始值，預設是 0.025，其餘比較不重要就不做說明。

另外我有用到 word2phrase，他能將其實意義一樣但因為符號或其他原因長得不太一樣的字處理成同一個字，e.g. Los Angeles 跟 Los\_Angelos 長得不一樣，但都是洛杉磯

## 2.2. 將 word2vec 的結果投影到 2 維的圖：

答：(圖)



2.3. 從上題視覺化的圖中觀察到了什麼？

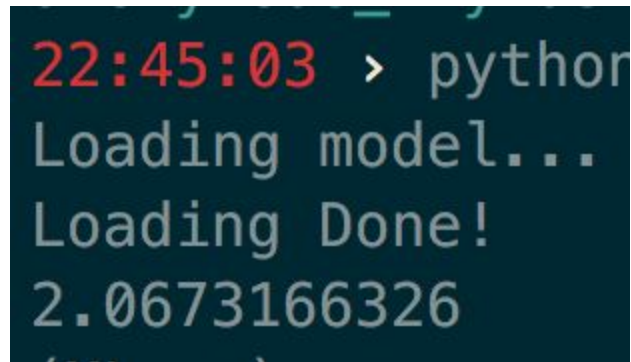
答：我發現同一種類的字會落在相近的位置。像是左下角有一區都是人名；右下角那一區是書名。而中間偏左的部分是動詞，而裡面又可以細分成現在式的動詞跟過去式的動詞；而中間偏右的部分有身體部位。

3.1. 請詳加解釋你估計原始維度的原理、合理性，這方法的通用性如何？

答：先說明我的訓練方法再來解釋原理。首先我照著順序產生 1 ~ 60 的 true dimension，並隨機產生他們各自的 hidden layer dimension，過 gen.py 產生屬於各自 true dimension 的一大堆測資。在每一個 true dimension 下，會有 r 輪產生一筆測資的行為，在每一輪中我會 sample 50 個點出來計算他們與第一個鄰居的距離的平均，最後把這些平均再根據 r 平均一次取得這個 true dimension 的代表距離。最後存成一張表，在 test 的時候就用跟在 training 時一樣算距離的方式算出那一筆 test data 的代表距離，並用之來查表反查出有著最相近距離時的 true dimension 是多少。原理是因為 data 變異很小，隨便 sample 出來的點與他最近的距離的變異不會太大，而此距離本身會因為維度上升而嚴格遞增，所以可以用來最為辨識的標準之一。通用性的話我覺得非常低，會 work 只是因為 gen.py 產生的 data 變異很低，所以使用他產生的測資做出來的預估還蠻準確的。

3.2. 將你的方法做在 hand rotation sequence dataset 上得到什麼結果？合理嗎？請討論之。

答：



```
22:45:03 > python
Loading model...
Loading Done!
2.0673166326
```

上圖是我跑出來的結果，預測出來接近二維。由於有參考一篇 MLE 的 paper (<https://papers.nips.cc/paper/2577-maximum-likelihood-estimation-of-intrinsic-dimension.pdf>)，裡面提到 hand rotation 的維度很有可能是三維左右，因為圖本身是二維，旋轉面又是一維。儘管我預測出結果與三維差距只有一個維度，但由於本身維度很小，所以算是很不準。我預測出來的結果不合理其實蠻正常的，因為我的模型本身通用性就很差，然後為了要能帶入 hand rotation 的圖片，我還用 scipy.misc 做了 imresize 的動作，所以圖片可以說是壓縮了，當然預測的就會更不好。這樣做出來會是二維感覺就只是他知道這是一張圖片，但內容已經辨識不出來了。