

고급 소프트웨어 실습 - 9주차 Face Recognition

2반

20140938 임다은

숙제: Eigenvector의 개수와 인식률의 상관 관계

투영에 사용되는 eigenvector의 개수를 나타내는 c 값을 {50, 100, 200, 300}와 같이 변화시키면서 각각의 인식률을 구해보자. 상위 몇 개의 eigenvector를 제거하고도 인식률을 구해본다.

코드에서 c 값을 나타내는 변수는 `ndim_reduced`이다. 이 `ndim_reduced`를 문제에 명시된 대로 조작하여 실험을 진행해보면 다음과같은 결과가 나온다.

1) $c = 50$

```
Console Window Started:

Training is completed...
ndim = 1764, nGallery = 350
acc = 70.0 (%)
ndim = 1764, nGallery = 350
acc = 62.0 (%)
```

2) $c = 100$

```
Console Window Started:

Training is completed...
ndim = 1764, nGallery = 350
acc = 72.0 (%)
ndim = 1764, nGallery = 350
acc = 66.0 (%)
```

3) $c = 200$

```
Console Window Started:

Training is completed...
ndim = 1764, nGallery = 350
acc = 72.0 (%)
ndim = 1764, nGallery = 350
acc = 68.0 (%)
```

4) $c = 300$

```
Console Window Started:

Training is completed...
ndim = 1764, nGallery = 350
acc = 72.0 (%)
ndim = 1764, nGallery = 350
acc = 64.0 (%)
```

c 가 200인 실험까지는 지속적으로 인식율이 증가하다가, c 가 300이 넘어가면 너무 노이즈가 심해져서 오히려 인식율이 저하되는 것을 알 수 있다.

상위 몇 개의 eigenvector를 지우기 위해 정렬이 내림차순인지 오름차순인지를 먼저 확인했어야 했다. 이를 확인하기 위해 OnFacerecognitionTrain() 함수에서 E 의 값들을 찍어본 eigenvalue들이 결과 내림차순으로 정렬되어있음을 알 수 있었다.

```
Console Window Started:

E:
961780352.000000 328782176.000000 112854256.000000 64963144.000000 48474964.0000
00 39736396.000000 38183712.000000 30037156.000000 25041052.000000 20646324.0000
00
Training is completed...
```

따라서 가장 큰 값인 첫 값들 몇 개를 제거하기 위하여

Mat $V_{\text{reduced}}(V, \text{Rect}(\text{ndim_start}, 0, \text{ndim_reduced}-\text{ndim_start}, \text{ndim}))$ 코드를 조작하였다. 이 곳에서 ndim_start 의 값을 조정하여 들어갈 상위 eigenvector를 제거해보았다. 이를 위해 ndim_reduced 와 함께 선언된 ndim_start 의 값을 조정해보았다. 다음 그림은 그 결과들이다.

1) $\text{ndim_reduced} = 100, \text{ndim_start} = 10$

```
PCAfaceRecognition.AppID.NoVersion
Console Window Started:

Training is completed...
ndim = 1764, nGallery = 350
acc = 72.0 (%)
ndim = 1764, nGallery = 350
acc = 62.0 (%)
```

ndim_Start 가 0인 원래 실험값에 비해서 2번째 test가 66%에서 62%로 감소했음을 알 수 있다.

2) ndim_reduced = 100, ndim_start = 50

```
Console Window Started:

E:
961780352.000000 328782176.000000 112854256.000000 64963144.000000 48474964.0000
00 39736396.000000 38183712.000000 30037156.000000 25041052.000000 20646324.0000
00
Training is completed...
ndim = 1764, nGallery = 350
acc = 68.0 (%)
ndim = 1764, nGallery = 350
acc = 58.0 (%)
```

배제하는 eigenvector가 늘어나면 정확도가 훨씬 더 많이 떨어지는 것을 알 수 있다.

ndim_start가 0인 경우 72, 66 와 ndim_start 10인 경우 72, 62 값을 비교하면 확연한 차이다.

3) ndim_reduced = 200, ndim_start = 100

```
Console Window Started:

E:
961780352.000000 328782176.000000 112854256.000000 64963144.000000 48474964.0000
00 39736396.000000 38183712.000000 30037156.000000 25041052.000000 20646324.0000
00
Training is completed...
ndim = 1764, nGallery = 350
acc = 70.0 (%)
ndim = 1764, nGallery = 350
acc = 66.0 (%)
```

4) ndim_reduced = 200, ndim_start = 150

```
Console Window Started:

E:
961780352.000000 328782176.000000 112854256.000000 64963144.000000 48474964.0000
00 39736396.000000 38183712.000000 30037156.000000 25041052.000000 20646324.0000
00
Training is completed...
ndim = 1764, nGallery = 350
acc = 70.0 (%)
ndim = 1764, nGallery = 350
acc = 60.0 (%)
```

3,4의 두 경우도 ndim_start가 0일 때 72, 68이 나온 것과 비교하면 ndim_start가 늘어날수록, 값이 큰 eigenvector들을 배제할수록 정확도가 떨어지는 것을 볼 수 있다.