作业六

关文聪 2016060601008

1 使用 PCA 对 Yale 人脸数据集进行降维,并分别观察前 20、前 100 个特征向量所对应的图像。请随机选取 3 张照片来对比效果。数据集 http://vision.ucsd.edu/content/yale-face-database

下载数据集并解压,先设置图片路径,根据路径读取图片的人脸数据,使用矩阵存储读取的数据。设定要保留的特征数量 k 为 20、100,调用 PCA 函数进行主成分分析(PCA)。将得到的结果再输出为图像,与原图进行比较。

Python 代码:

```
import numpy as np
import scipy.misc as misc
import matplotlib.pyplot as plt
from sklearn.decomposition import PCA
import os
```

数据的读取与初始化预处理

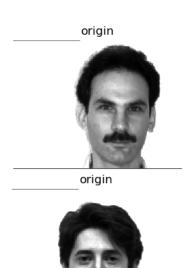
```
path = 'C:\\Users\\Eternity-Myth\\Desktop\\yalefaces'
for dirpath, subdir, file_set in os.walk(path):
    all_img = [path + '\\' + f for f in file_set] # 保存所有文件的路径

m, n = len(all_img), len(misc.imread(all_img[0]).ravel()) # 行和列的数据
data = np.zeros((m, n)) # 初始化数据为(m,n)形状的矩阵
for i, f in enumerate(all_img):
    img = misc.imread(f).ravel() # 将每个 2D 图像展平为 1D 阵列
    data[i] = img
```

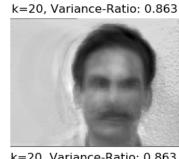
```
# 对数据进行主成分分析 (PCA) 处理
data centered = data - data.mean(axis=0) # 对所有数据进行中心化
data_centered -= data_centered.mean(axis=1).reshape(m, -1) # 对所有参数进行中心化
gap = data - data centered # 保存数据与中心化处理后的数据之间的关系
k = [20, 100] # 保留的特征数 k, 设定 k 为 20 与 100
pca1, pca2 = PCA(n components=k[0]), PCA(n components=k[1])
r set, im set = [], [] # 保存每个 pca 的方差比, 输出去中心 1D 数组
for pca in [pca1, pca2]:
   lower_data = pca.fit_transform(data_centered) # 形状是(166, k)
   comp = pca. components_ # 形状是(k, 77760), 这是一个稀疏的二维数组
   r set.append(np. sum(pca. explained variance ratio))
   im set.append(np.dot(lower data, comp) + gap)
# 输出处理过后的数据图像
for j in range (1, 166):
   # 原图
   fig, [ax0, ax1, ax2] = plt. subplots(1, 3, figsize=(10, 2.2))
   ax0.imshow(data[j].reshape((243, 320)), cmap=plt.cm.gray)
   ax0. set title ('primal')
   ax0. axis ('off')
   # PCA 降维后的图像
   for i, ax in enumerate([ax1, ax2]):
       ax.imshow(im_set[i][j].reshape((243, 320)), cmap=plt.cm.gray)
       ax.set title('k=%s, Variance-Ratio: %.3f' % (k[i], r_set[i]))
       ax. axis ('off')
   plt. subplots adjust (left=0.02, bottom=0.05, right=0.98, wspace=0)
    plt.savefig(r'C:\\Users\\Eternity-Myth\\Desktop\\output\\' + str(j) + '.png')
```

output					▼ ひ 捜索"output" メ
9 9 9	9 9 9	999 10 10 1	9 9 9 9 9		999 999 999
0 0 0	2	3 4	5 6	7 8	9 10 11
12	13	14 15	16 17	18 19	20 21 22
9 9 9					9 9 9 9 9 9 9
23	24	25 26	27 28	29 30	31 32 33
34	35	36 37	38 39	40 41	42 43 44
9 9 9	and the same of th				9999999
45	46	47 48	49 50	51 52	53 54 55
2 2 2	0 0 0				0 0 0 0 0 0 0 0
56	57	58 59	60 61	62 63	64 65 66
	0 0 0		999999		9 0 0 0 0 0 0 0
67	68	69 70	71 72	73 74	75 76 77
0 0 0	0 0 0		00000		
78	9 9 9	80 81	82 83	84 85	9 9 9 9 9 9 88
Q Q Q	9 9 9	ର ର ର ଆ ଆ ଆ	99999	10 10 10 2 2 2	ରୁଜୁଜୁଜୁଜୁଜୁ
89	90	91 92	93 94	95 96	97 98 99
0.0.0	0 0 0				
100	101	102 103	104 105	106 107	Q Q Q Q Q Q Q 110
0 0 0	0 0 0			00000	
111	112	113 114	115 116	117 118	119 120 121
122	123	124 125	126 127	128 129	130 131 132
0 0 0	9 9 9	9 9 9 9 9 9	9 9 9 9 9		9 9 9 9 9 9 9
133	134	135 136	137 138	139 140	141 142 143
9 9 9	9 9 9	9 0 0 9 9 9	140	150 151	0 0 0 0 0 0 0
144	145	146 147	148 149	150 151	152 153 154
155	156	157 158	159 160	161 162	163 164 165

如图所示,在设置的 output 文件夹下已经生成了对应的输出图像。随机选取 3 张图片对比效果如下:

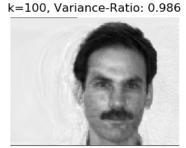
















由对比结果可见,选取 PCA 的特征数越多,图像越清晰、明显,而在特征数不足时,会出现大量阴影轮廓,可以用 pca. explained_variance_ratio_来查看当前选择的最大 k 个特征向量的方差占比,方差占比越大则此特征表征的信息越多。可以发现当 k 从 20 增加到 100 时,选择的 k 个特征的累计方差占比已经接近于 1 了,而相应地,图像的特征已经与原始图像非常接近了。这意味着我们可以用大约 100 维的向量来描述一张原本维数达数万维的图像,可见 PCA 在这样的灰度人脸图像下的降维是非常有效的。