《数据科学与工程算法基础》实践报告

报告题目： 基于 PCA 算法对图像压缩处理

姓　　名： 杨东东

学　　号： 51194507017

完成日期： 2020-08-03

摘要：

图像的压缩处理是一种很常见的场景，可以在保留重要信息的情况下占用更少的存储空间或传输带宽。本文使用 PCA 算法，又叫主成分分析，对一批航拍图像进行压缩。最后，对压缩效果做一些简单的统计评估。

Abstract:

Image compression processing is a very common scenario, which can occupy less storage space or transmission bandwidth while retaining important information. In this paper, a batch of aerial images are compressed using the familiar PCA algorithm, also called principal component analysis. Finally, a simple statistical evaluation of the compression effect is made.

1. 背景

随着多媒体和通信技术的快速发展，多媒体信息的传输对数据的存储和传输提出了更高的要求，也给现有的有限的带宽以严峻的考验，特别是具有庞大数据量的数字图像通信，更难以传输和存储。图像压缩的目的就是把原来较大的图像用尽量少的字节来表示和传输，并且要求压缩后的图像有较好的质量。

本文采用 PCA 算法来对图像进行压缩处理，图像来自无人机航拍，共计 650 张。

2. 问题描述

主成分分析（Principal Component Analysis，PCA）， 是一种统计方法。通过正交变换将一组可能存在相关性的变量转换为一组线性不相关的变量，转换后的这组变量叫主成分。

在许多领域的研究与应用中，往往需要对反映事物的多个变量进行大量的观测，收集大量数据以便进行分析寻找规律。多变量大样本无疑会为研究和应用提供了丰富的信息，但也在一定程度上增加了数据采集的工作量，更重要的是在多数情况下，许多变量之间可能存在相关性，从而增加了问题分析的复杂性，同时对分析带来不便。如果分别对每个指标进行分析，分析往往是孤立的，而不是综合的。盲目减少指标会损失很多信息，容易产生错误的结论。

因此需要找到一个合理的方法，在减少需要分析的指标同时，尽量减少原指标包含信息的损失，以达到对所收集数据进行全面分析的目的。由于各变量间存在一定的相关关系，因此有可能用较少的综合指标分别综合存在于各变量中的各类信息。主成分分析就属于这类降维的方法。

3. 方法

3.1 统一图像尺寸

图像数据集的尺寸是大小不一的，首先把图像尺寸统一标准化。这里有两种思路：一是图像裁剪，在每张图像上取相同长宽窗口并保存，丢弃窗口之外的部分；二是图像缩放，将多个输入像素映射为一个输出像素。

图 3-1 001.jpg

图 3-2 001-裁剪.jpg

图 3-3 001-放缩.jpg

本文选择的是图像放缩，借助 python 的 PIL 库中的最近滤波方法来处理图像，统一后的图像为 512*512。

3.2 PCA 的实现

PCA 算法的实现一般有两种：基于特征值分解的协方差矩阵的 PCA 实现、基于 SVD 分解协方差矩阵的 PCA 实现。本文选择前者，具体步骤如下：

1）分别求每个维度的平均值，然后对于所有的样例，都减去对应维度的均值，得到去中心化的数据。这一步非必要，对于去中心化的数据和不去时候的数据，特征值大小会不同，但是相对大小并不会改变；

2）求协方差矩阵 C：$\frac{1}{N-1}XX^T$，用去中心化的数据矩阵乘上它的转置，然后除以(N-1)即可，N 为样本数量，此处指图像像素矩阵的行；

3）求协方差的特征值和特征向量，$\frac{1}{N-1}XX^T$是方阵，很方便求解；

4）将特征值按照从大到小排序，选择前 k 个，然后将其对应的 k 个特征向量分别作为列向量组成特征向量矩阵 P。

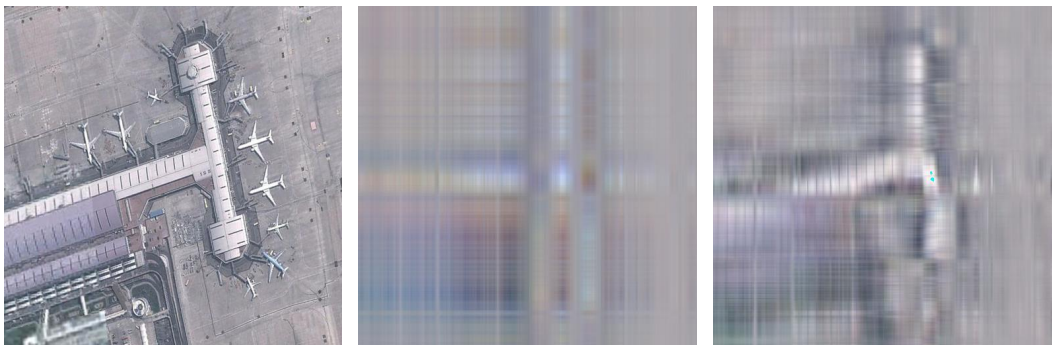
5）将样本点从原来维度投影到选取的 k 个特征向量，得到低维数据；

6）通过逆变换，重构低维数据，进行复原；

7）将处理后的图像写入新目录。

4. 实验结果

PCA 处理后的效果：



(a) Original　　(b) *K*= 1　　(c) *K*= 4

(d) *K*= 8　　(e) *K*= 16　　(e) *K*= 64

图 4-1 不同 K 值的压缩对比

以 011.jpg 为例，原图是一个机场俯视图。当 K=1 时，图像是几种颜色的横竖拉伸；K=4 和 8 时，可以看到白色建筑的形状；K=16，飞机轮廓清晰；K=64 时候，大部分信息都能显示出来，局部仍有噪音。

4.1 信息丢失率

定义 PCA 的误差计算公式：

$$\text{error} = \frac{\sum(data_i - recdata_i)^2}{\sum data_i^{\,2}} \times 100\%, \qquad \text{i} = 0,1 \ldots \text{n} - 1.$$

其中$data_i$是原图像矩阵的行向量，$recdata_i$是对应 PCA 压缩后图像的行向量。

表 1：不同 K 值的信息丢失率

K	$error_{min}(\%)$	$error_{max}(\%)$	$error_{avg}(\%)$
1	0.236	145.23	12.83
4	0.168	143.72	9.51
8	0.126	124.46	7.72
16	0.080	109.70	5.92
32	0.041	91.15	4.25
64	0.018	62.75	2.76
128	0.007	26.92	1.39

主成分个数保留越多，压缩后图像和原图像之间的信息差值越小，这一变化趋势也符合直观的图像显示效果。当 K=64 时，平均每张压缩图像与原始图像的误差是 2.762%。

4.1 空间压缩率

定义表示空间压缩率的计算公式：

$$\text{c} = \frac{Size_{compressed}}{Size_{original}} \times 100\%$$

其中$Size_{compressed}$是压缩后图像的所占空间的大小，$Size_{original}$是指原始图像。

计算不同 K 值的每张图像的空间压缩率极值、平均值，如表 2 所示。随着 K 值的设定增长，图像的空间压缩率也随之下降，即主成分保留越多，压缩图像越接近原始图像大小，反之则压缩效果愈明显。

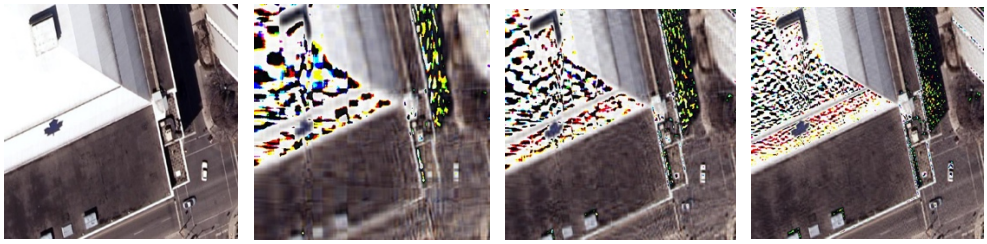
表 2：不同 K 值的空间压缩率

K	$c_{min}(\%)$	$c_{max}(\%)$	$c_{avg}(\%)$
1	20.78	88.89	35.00
4	30.43	143.61	49.28
8	38.29	182.31	59.98
16	48.57	227.14	72.89
32	64.44	268.15	86.49
64	80.14	284.22	98.45
128	94.10	219.22	105.24

5. 结论

本文使用 PCA 算法对图像进行压缩处理，包括图像显示效果、信息丢失率、空间压缩率是符合预期的。

在统计存储空间变化时，发现有几张图像的压缩效果非常差，几乎是原图像大小的 2~3 倍，进一步查找，这些 bad case 都集中在 382.jpg、383.jpg、414.jpg 这 3 张图像。



(a) 32KB (b) 54KB(K=16) (c) 65KB(K=32) (d) 73KB(K=64)

图 5-1 382.jpg 的几种压缩效果

通过观察发现，382.jpg、383.jpg 这两张图像有相当大比例的白色区域，如图 4-2 中的左上角，414.jpg 有大面积的黑色区域。结合压缩后的图片推理：在[0,255]两端的像素集中在某区域时，PCA 压缩后更容易产生噪声点，这些噪声点保存在图象中会比原图像占用更大的存储空间。

最后，如何选取合适的 K 值策略，使得图像的信息丢失和节约存储空间之间达到一种平衡，是值得进一步探索的方向。