

# 统计分析实验三：利用PCA实现图像压缩

16337292 袁浩扬

## 问题描述

输入一张灰度图片Lena，使用PCA方法把原始图片分别按照2:1、8:1、32:1进行压缩，即压缩后的数据量为原始图片的1/2、1/8、1/32。分析压缩后的数据所含信息量大小，并比较压缩数据再经过重建后与原始图片的视觉差异。

## PCA（主成分分析）

PCA 的目标是寻找  $r$  ( $r < n$ ) 个新变量，使它们反映事物的主要特征，压缩原有数据矩阵的规模，将特征向量的维数降低，挑选出最少的维数来概括最重要特征。

每个新变量是原有变量的线性组合，体现原有变量的综合效果，具有一定的实际含义。这  $r$  个新变量称为“主成分”，它们可以在很大程度上反映原来  $n$  个变量的影响，并且这些新变量是互不相关的，也是正交的。

通过主成分分析，压缩数据空间，将多元数据的特征在低维空间里直观地表示出来。

## 算法描述

- 计算矩阵  $X$  的样本的协方差矩阵  $S$
- 计算协方差矩阵  $S$  的特征向量  $e_1, e_2, \dots, e_n$  和特征值  $t_1, t_2, \dots, t_n$ ,  $t = 1, 2, \dots, n$
- 投影数据到特征向量张成的空间之中。利用公式  $\text{newBV} = \sum e_i B V$ ，其中  $BV$  为原样本中对应维度的值。

## Matlab实现

- 定义函数 `pac(P)`， $P$  为传入的压缩率

```
function pac(P)
% P为压缩率
```

- 输入图像，定义每个子块的尺寸，由题得，图像尺寸为  $512 \times 512$ ，此时得到了  $512 \times 512 / (16 \times 16) = 1024$  个样本，每个样本初始维度为  $16 \times 16 = 256$  维。

```
img = imread('原始图片.bmp');
figure(1), subplot(131), imshow(img, []);
title('Original Image');
[M N] = size(img);
block_size = 16; %子块的尺寸
original_W = block_size * block_size; %初始维度
end_W = P * original_W; %压缩后的维度
```

- 开始PCA，首先将图像矩阵转换为列矩阵，其中每一列为一个样本，故此时列矩阵size应为  $256 \times 1024$ 。

```
% PCA
colMat = im2col(double(img), [block_size block_size], 'distinct'); %将图像块转为列向量
```

- 计算每个样本（每列）的灰度均值，列矩阵元素减去对应均值，即对列矩阵进行白化处理。

```
mean_ = ones(size(colMat,1),1) * mean(colMat); %计算每块的灰度均值
colMat = colMat - mean_; %白化
```

- 计算每个协方差矩阵，对于样本的协方差矩阵，有公式  $MM' / (Weight - 1)$ ，其中  $M$  为样本矩阵， $Weight$  为每个样本的维度。

```
covarianceMat = colMat * colMat' / (size(colMat,2) - 1); %计算协方差矩阵
```

- 得到协方差矩阵的特征向量以及特征值，并以特征值为key，获取降序的特征向量矩阵  $E$ 。

```
[E,D] = eig(covarianceMat); %E为特征向量，D为特征值
[temp,order] = sort(diag(D),'descend');
E = E(:,order); %按特征值降序排列
```

- 根据要求的压缩后的维度 `end_W`，得到前 `end_W` 个特征向量，并对图像进行复原。

```
E_leave = E(:,1:end_W);
g_proj = colMat' * E_leave;
g_rec = g_proj * E_leave';
```

8. 每个压缩后的样本加上对应的均值，复原灰度并得到复原后的图像矩阵。

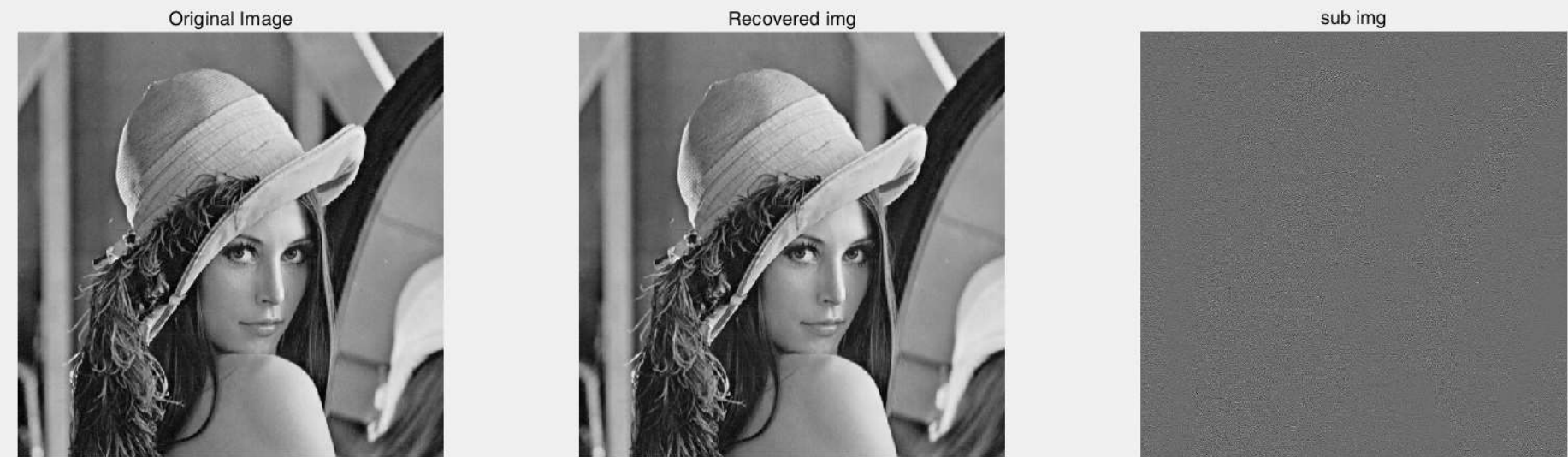
```
s = g_rec' + mean_  
s = col2im(s, [block_size block_size], [M N], 'distinct');
```

9. 显示复原后的图像以及复原前后图像的插值。

```
figure(1),subplot(132),imshow(s,[]);  
title('Recovered img');  
  
sub = double(img) - s;  
figure(1),subplot(133),imshow(sub,[]);  
title('sub img');
```

# 实验结果

1. 压缩比例为1/2的图像。 `pca(1/2)`



2. 压缩比例为1/8的图像。 `pca(1/8)`



3. 压缩比例为1/32的图像。 `pca(1/32)`

