# DIP Lab 4: Image Compression

2017011620　计73　李家昊

2020 年 6 月 3 日

## 1　实验一

### 1.1　实验原理

本实验将实现离散余弦变换（DCT）及其逆变换（IDCT），其中DCT将信号从时域映射到频域，而IDCT将信号从频域映射到时域。

对于一维DCT，给定一个$N$维向量$\mathbf{x}$，其$N$点DCT为$\mathbf{y}$，则有，

$$y_k = \alpha_k^{(N)} \sum_{n=0}^{N-1} x_n \cos\left(\frac{(2n+1)k\pi}{2N}\right), \quad k = 0, 1, 2, \cdots, N-1 \tag{1}$$

其中，

$$\alpha_k^{(N)} = \begin{cases} \sqrt{1/N}, & k = 0 \\ \sqrt{2/N}, & k \geq 1 \end{cases} \tag{2}$$

记矩阵$\mathbf{A}_N = (a_{kn}^{(N)}) \in \mathbb{R}^{N\times N}$，其中，

$$a_{kn}^{(N)} = \alpha_k^{(N)} \cos\left(\frac{(2n+1)k\pi}{2N}\right), \quad k, n = 0, 1, 2, \cdots, N-1 \tag{3}$$

则一维DCT可以表示成矩阵形式，直接计算的时间复杂度为$O(N^2)$，

$$\mathbf{y} = \mathbf{A}_N \mathbf{x} \tag{4}$$

对于二维DCT，给定一个$M \times N$矩阵$\mathbf{X}$，其二维DCT为$\mathbf{Y}$，则有，

$$Y_{uv} = \alpha_u^{(M)} \alpha_v^{(N)} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} X_{xy} \cos\left(\frac{(2x+1)u\pi}{2M}\right) \cos\left(\frac{(2y+1)v\pi}{2N}\right) \tag{5}$$

将方程 (5)稍作变换可知，二维DCT可以看作两次一维DCT，

$$Y_{uv} = \alpha_v^{(N)} \sum_{y=0}^{N-1} \left\{ \cos\left(\frac{(2y+1)v\pi}{2N}\right) \left[ \alpha_u^{(M)} \sum_{x=0}^{M-1} X_{xy} \cos\left(\frac{(2x+1)u\pi}{2M}\right) \right] \right\} \tag{6}$$

同样有$\mathbf{A}_M = (a_{ux}^{(M)}) \in \mathbb{R}^{M\times M}$，$\mathbf{A}_N = (a_{vy}^{(N)}) \in \mathbb{R}^{N\times N}$，根据方程 (6)，首先对每一列做一维DCT，即$\mathbf{A}_M\mathbf{X}$，再对每一行做一维DCT，即$(\mathbf{A}_N(\mathbf{A}_M\mathbf{X})^T)^T$，这样就得到了二维DCT的矩阵形式，计算的时间复杂度为$O(MN(M+N))$，

$$\mathbf{Y} = \mathbf{A}_M\mathbf{X}\mathbf{A}_N^T \tag{7}$$

容易验证$\mathbf{A}_N$为正交矩阵，由此可得一维和二维IDCT分别为，

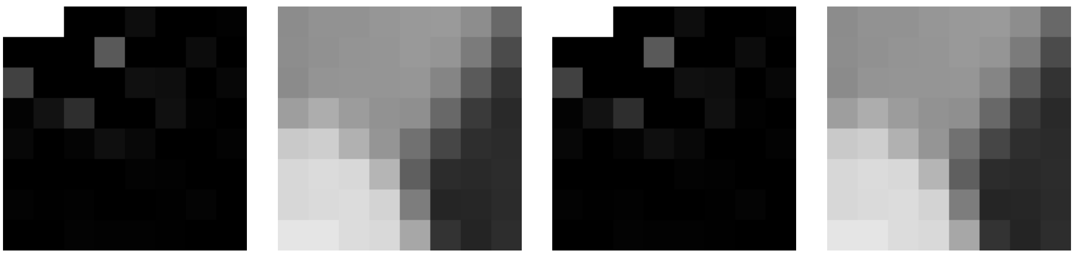
$$\mathbf{x} = \mathbf{A}_N^T\mathbf{y}, \quad \mathbf{X} = \mathbf{A}_M^T\mathbf{Y}\mathbf{X}_N \tag{8}$$

## 1.2 实验结果及分析

首先选取一个$8\times 8$区域作为例子，如图 1，对该区域分别进行二维DCT和两次一维DCT，频域图像和恢复的图像如图 2。经过计算，两种DCT方式处理的图片经过IDCT还原后，信噪比均为$+\infty$，说明两种处理方式是等价的，这也与实验原理部分的理论分析相符。

(a) 帽檐处白框内的$8\times 8$区域　　(b) 放大的$8\times 8$区域

图 1: 选取女生帽檐处白框内的$8\times 8$区域



(a) DCT2D　　(b) DCT2D-IDCT　　(c) DCT1Dx2　　(d) DCT1Dx2-IDCT

图 2: 在$8\times 8$区域内二维DCT和两次一维DCT的对比结果

接下来将原图分割为若干个$8\times 8$区域，每个区域内分别用二维DCT和两次一维DCT处理，得到频域图像后，仅保留左上角低频部分的系数，去掉高频部

分，在不同的保留比例下，得到的结果如图 3，对应的峰值信噪比（PSNR）如表 1。实验结果又一次验证了，二维DCT和两次一维DCT的处理效果是完全相同的。同时也可以看出，图像的大部分信息主要集中在低频部分，当保留比例为1/64时，得到的结果依然是可辨认的，如图 3(d)。

(a) 保留所有系数 (b) 保留1/4系数 (c) 保留1/16系数 (d) 保留1/64系数

图 3: 保留不同比例的频域系数的对比结果

表 1: 在不同的系数保留比例下，复原图像的峰值信噪比PSNR

| Method \ #Coef. | 1/1 | 1/4 | 1/16 | 1/64 |
|---|---|---|---|---|
| DCT2D | 67.39 | 32.95 | 26.41 | 22.04 |
| DCT1Dx2 | 67.39 | 32.95 | 26.41 | 22.04 |

调整方块区域的大小，再次进行实验，实验结果如表 2。可以看出，方块区域越大，保留比例越低，那么丢失的信息就越多，block effect就越明显，PSNR也就越低。

表 2: 在不同的方块大小以及系数保留比例下，复原图像的峰值信噪比PSNR

| Block \ #Coef. | 1/1 | 1/4 | 1/16 | 1/64 | 1/256 | 1/1024 |
|---|---|---|---|---|---|---|
| $1 \times 1$ | $+\infty$ | | | | | |
| $2 \times 2$ | 74.23 | 29.95 | | | | |
| $4 \times 4$ | 70.58 | 31.62 | 25.35 | | | |
| $8 \times 8$ | 67.39 | 32.95 | 26.41 | 22.04 | | |
| $16 \times 16$ | 62.60 | 33.64 | 27.33 | 22.93 | 19.36 | |
| $32 \times 32$ | 60.61 | 33.93 | 27.69 | 23.51 | 20.09 | 17.15 |

# 2 实验二

## 2.1 实验原理

实验一通过直接去掉某些高频系数来压缩图像，这往往会带来一些block effect。为了缓解这一问题，实验二引入了量化表来过滤高频，这也是JPEG压缩格式的做法。在一个$8\times 8$方块内，记量化表为$Q$，原始图像通过DCT得到频谱为$\mathbf{Y}$，则量化后的结果为，

$$Y_q(u,v)=\mathrm{round}\left(\frac{Y(u,v)}{Q(u,v)}\right) \tag{9}$$

量化完成后，再通过编码，就生成了压缩后的图像数据。

恢复图像时，首先解码，然后反量化，

$$Y(u,v)=Y_q(u,v)\cdot Q(u,v) \tag{10}$$

再进行IDCT，就可以恢复图像了。

量化表有很多种设计，本次实验主要对比三种量化表，分别是Canon DIGITAL IXUS 60 (fine)，记为$Q_1$，

$$Q_1=\begin{pmatrix} 1 & 1 & 1 & 2 & 3 & 6 & 8 & 10 \\ 1 & 1 & 2 & 3 & 4 & 8 & 9 & 8 \\ 2 & 2 & 2 & 3 & 6 & 8 & 10 & 8 \\ 2 & 2 & 3 & 4 & 7 & 12 & 11 & 9 \\ 3 & 3 & 8 & 11 & 10 & 16 & 15 & 11 \\ 3 & 5 & 8 & 10 & 12 & 15 & 16 & 13 \\ 7 & 10 & 11 & 12 & 15 & 17 & 17 & 14 \\ 14 & 13 & 13 & 15 & 15 & 14 & 14 & 14 \end{pmatrix} \tag{11}$$

Nikon CoolPix L12 (fine)，记为$Q_2$，

$$Q_2=\begin{pmatrix} 2 & 1 & 1 & 2 & 3 & 5 & 6 & 7 \\ 1 & 1 & 2 & 2 & 3 & 7 & 7 & 7 \\ 2 & 2 & 2 & 3 & 5 & 7 & 8 & 7 \\ 2 & 2 & 3 & 3 & 6 & 10 & 10 & 7 \\ 2 & 3 & 4 & 7 & 8 & 13 & 12 & 9 \\ 3 & 4 & 7 & 8 & 10 & 12 & 14 & 11 \\ 6 & 8 & 9 & 10 & 12 & 15 & 14 & 12 \\ 9 & 11 & 11 & 12 & 13 & 12 & 12 & 12 \end{pmatrix} \tag{12}$$

以及Jpeg Standard，记为$Q_3$，

$$Q_3 = \begin{pmatrix} 16 & 11 & 10 & 16 & 24 & 40 & 51 & 61 \\ 12 & 12 & 14 & 19 & 26 & 58 & 60 & 55 \\ 14 & 13 & 16 & 24 & 40 & 57 & 69 & 56 \\ 14 & 17 & 22 & 29 & 51 & 87 & 80 & 62 \\ 18 & 22 & 37 & 56 & 68 & 109 & 103 & 77 \\ 24 & 35 & 55 & 64 & 81 & 104 & 113 & 92 \\ 49 & 64 & 78 & 87 & 103 & 121 & 120 & 101 \\ 72 & 92 & 95 & 98 & 112 & 100 & 103 & 99 \end{pmatrix} \tag{13}$$

## 2.2 实验结果及分析

三种量化表处理后，复原图像如图 4，峰值信噪比如表 3。三张复原图像中，肉眼几乎看不出区别，从PSNR数值来看，Canon和Nikon差异不大，但两者均显著优于JPEG。原因可能是JPEG量化表的数值更大，压缩比更高，带来的数据流失也相对更多。

(a) Canon (b) Nikon (c) JPEG

图 4: 三种量化表的处理结果

表 3: 三种量化表处理后，复原图像的峰值信噪比PSNR

| | Canon | Nikon | JPEG |
|---|---|---|---|
| PSNR | 42.93 | 43.85 | 35.15 |

进一步考虑在量化表前添加一个系数$a$，以$aQ$来作量化处理，随着$a$的变化，PSNR的变化如图 5，压缩率变化如图 6。可以看到，随着$a$的增长，量化表的数值越大，图像压缩率越高，损失的信息越多，因此PSNR越低。压缩比和PSNR通常是矛盾的，在实际应用中，需要找到两者之间的平衡。

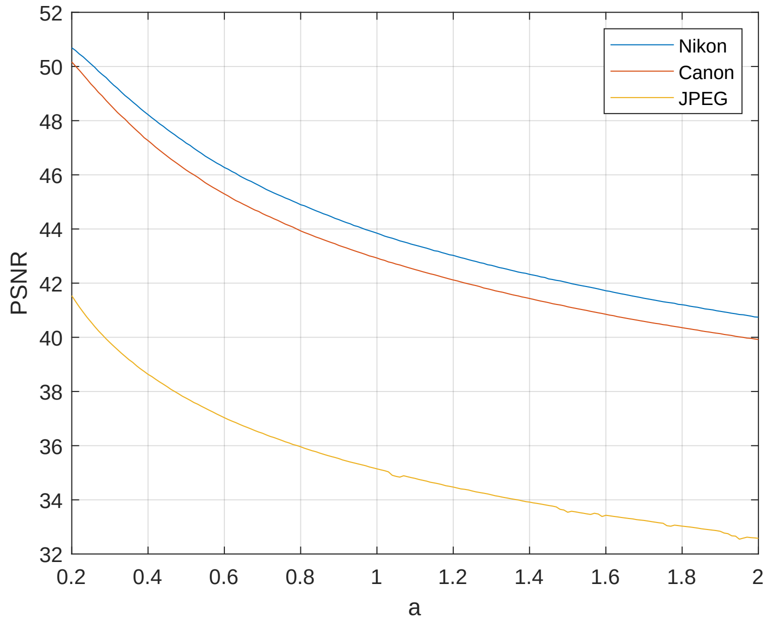



图 5: 随系数$a$的变化，量化表$aQ$的PSNR

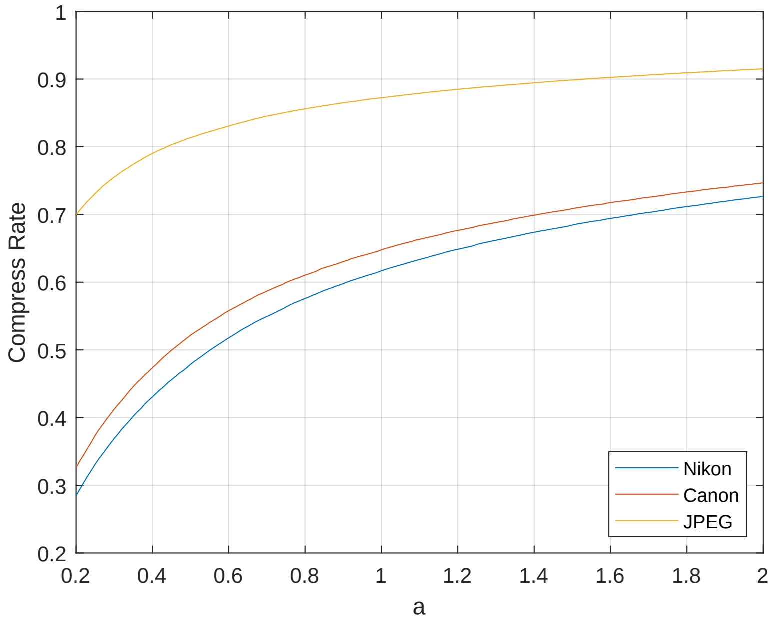



图 6: 随系数$a$的变化，量化表$aQ$的压缩率