Image Compression Report

郑凯文

2021 年 5 月 5 日

1 DCT

对于图像中$N \times N$大小的block，其DCT变换为

$$c(u,v) = \alpha(u)\alpha(v)\sum_{x=0}^{N-1}\sum_{y=0}^{N-1} f(x,y)\cos(\frac{(2x+1)u\pi}{2N})\cos(\frac{(2y+1)v\pi}{2N}), \quad u,v = 0,1,\ldots,N-1 \tag{1}$$

逆变换为

$$f(x,y) = \sum_{u=0}^{N-1}\sum_{v=0}^{N-1} \alpha(u)\alpha(v)c(u,v)\cos(\frac{(2x+1)u\pi}{2N})\cos(\frac{(2y+1)v\pi}{2N}), \quad x,y = 0,1,\ldots,N-1 \tag{2}$$

其中

$$\alpha(a) = \begin{cases} \sqrt{1/N}, & a = 0 \\ \sqrt{2/N}, & a > 0 \end{cases} \tag{3}$$

若保留$1/k^2$的系数，则取矩阵$c(u,v)$左上角$N/k \times N/k$大小的区域，其余元素置0。在实现中，可以构造如下矩阵

$$A = \begin{bmatrix} \sqrt{1/N} & \sqrt{1/N} & \cdots & \sqrt{1/N} \\ \sqrt{2/N}\cos(\pi/2N) & \sqrt{2/N}\cos(3\pi/2N) & \cdots & \sqrt{2/N}\cos((2N-1)\pi/2N) \\ \sqrt{2/N}\cos(2\pi/2N) & \sqrt{2/N}\cos(6\pi/2N) & \cdots & \sqrt{2/N}\cos(2(2N-1)\pi/2N) \\ \vdots & \vdots & \ddots & \vdots \\ \sqrt{2/N}\cos((N-1)\pi/2N) & \sqrt{2/N}\cos((N-1)3\pi/2N) & \cdots & \sqrt{2/N}\cos((N-1)(2N-1)\pi/2N) \end{bmatrix} \tag{4}$$

设图像矩阵为F，则正变换和逆变换可以表示为

$$C = AFA^T, \quad F = A^TCA \tag{5}$$

1.1 1D-DCT && 2D-DCT

经过实验可以发现，分别进行两次1D-DCT、两次系数裁剪和进行一次2D-DCT、一次系数裁剪的效果完全相同。使用上节的矩阵，这容易从理论上解释：

暂时不考虑系数裁剪，首先对每行进行1D-DCT，原图中第x行对应着结果中第u行，这样

$$c_1(u,v) = \alpha(v) \sum_{y=0}^{N-1} f(u,y) \cos(\frac{(2y+1)v\pi}{2N}), \quad u,v = 0,1,\dots,N-1 \tag{6}$$

对照发现，这恰好可以使用上节的矩阵A表示

$$C_1 = FA^T \tag{7}$$

同理，再对列进行1D-DCT

$$C_2 = AC_1 = AFA^T \tag{8}$$

因此$C_2 = C$。继续考虑系数裁剪，设

$$A = \begin{bmatrix} A_k \\ A_{nk} \end{bmatrix} \tag{9}$$

其中A_k为矩阵A的前N/k行。对于2D-DCT，有

$$C = \begin{bmatrix} A_k \\ A_{nk} \end{bmatrix} F \begin{bmatrix} A_k^T & A_{nk}^T \end{bmatrix} = \begin{bmatrix} A_kFA_k^T & - \\ - & - \end{bmatrix} \tag{10}$$

截取左上角$N/k \times N/k$的区域后为$C_k = A_kFA_k^T$。对于1D-DCT，先对行进行

$$C_1 = F \begin{bmatrix} A_k^T & A_{nk}^T \end{bmatrix} = \begin{bmatrix} FA_k^T & FA_{nk}^T \end{bmatrix} \tag{11}$$

截取前N/k列后为$C_{1k} = FA_k^T$。再对列进行

$$C_2 = AC_{1k} = AFA_k^T = \begin{bmatrix} A_kFA_k^T \\ - \end{bmatrix} \tag{12}$$

截取前N/k行后为$C_{2k} = A_kFA_k^T = C_k$。所以二者还是等价的。

1.2 Ratio of Coefficients && Block Size

为了方便块的划分，我首先将328x328大小的图像resize为512x512。我选取了1、1/4、1/16、1/64四种系数比例和8、32、128、512四种块大小，计算其PSNR如下：

表 1: 不同系数比例和块大小下的PSNR表

PSNR \ 块大小 / 系数比例	8	32	128	512
1	318.36	307.36	295.43	282.01
1/4	40.07	42.85	43.53	43.77
1/16	29.69	31.42	32.10	32.28
1/64	24.11	25.84	26.57	26.71

从中可以看出

- 系数比例的减小会使PSNR明显下降
- 系数全部保留时，块大小越小PSNR越高（此时其实可完全恢复原图，这是精度问题造成的）；反之，块大小增加会使PSNR有微小的改进，但总的来说影响不大

下面选取一些图片来从视觉角度分析二者的影响。

(a) 1 (b) 1/4 (c) 1/16 (d) 1/64

图 1: 块大小为8时，不同系数比例下的效果图

从图1中看出，随着保留系数的减小，图像越来越粗糙，块大小为8、保留系数为1/64时只保留了左上角的一个分量，使得每块内为纯色。从图2中可以看出，随着块大小的增大，图像更加平滑，但会出现类似于衍射条纹的模式。在保留系数非常小（1/64）、块大小较小时，则会有很严重的blocking artifacts，图像几乎无法使用（图3）。

(a) 8 (b) 32 (c) 128 (d) 512

图 2: 系数比例为1/16时，不同块大小下的效果图

(a) 8 (b) 32 (c) 128 (d) 512

图 3: 系数比例为1/64时，不同块大小下的效果图

(a) 0.5 (b) 1.0 (c) 1.5 (d) 2.0

图 4: jpeg量化矩阵不同a值下的效果图

2 Quantization

使用$Q' = aQ$作为量化矩阵，图4展示了使用jpeg时，不同a值下的效果图。随着a增大，变换后矩阵中的值越来越小，零元比例增大，高频信息消失更多，图像也越来越粗糙。在$a = 2.0$时，图像已经出现了比较明显的blocking artifacts。

选取三种量化矩阵，在$a \in (0.1, 2.0)$范围内作出PSNR随$\log a$变化的曲线，如图5所示。可以看出PSNR大致和$\log a$线性相关，且由于jpeg量化矩阵中的元素相对较大，其曲线整体位于另外两者下方。

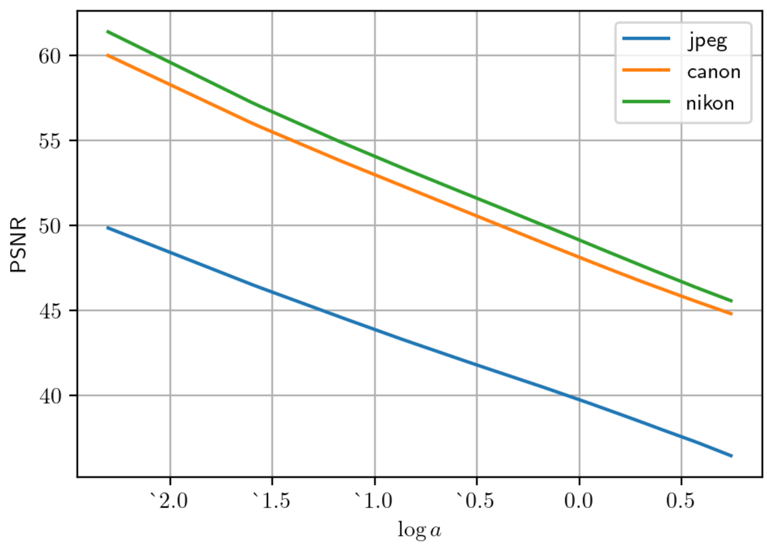

图 5: 三种量化矩阵下的PSNR-$\log a$曲线

关于如何从压缩率确定a，我搜索到jpeg标准中的做法How can I generalize the quantization matrix in JPEG compression?，其中说到一个叫品质因子的东西，其范围为1到100。给定品质因子Q，定义

$$S = \begin{cases} 5000/Q, & Q < 50 \\ 200 - 2Q, & Q \geq 50 \end{cases} \tag{13}$$

设原先量化矩阵为T，则使用的新量化矩阵为$T' = floor((ST + 50)/100)$，因此可以近似认为

$$a = \begin{cases} 50/Q, & Q < 50 \\ 2 - Q/50, & Q \geq 50 \end{cases} \tag{14}$$

从压缩率确定品质因子，则可以通过统计经验来推断。