

数字图像处理实验二

DCT变换

Experiment 2: DCT transform

姚鸿勋 刘绍辉
shliu@hit.edu.cn

哈尔滨工业大学计算机科学与技术学院
2018秋

实验内容

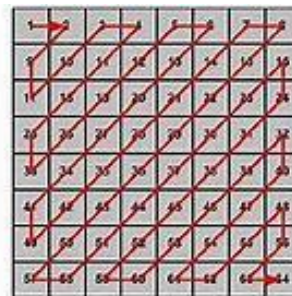


实验二

- 熟悉图像变换的思想：1维： $Y = FX$; 2维： $Y = FXF'$
- 编程实现二维8*8图像块的DFT, DCT变换
 - 输入一副RGB图像，将其转化为YCbCr颜色空间，然后对Y分量做分块DFT变换，将变换后的结果的幅度和相位分别显示出来并存储为灰度图像，统计计算时间
 - 如果分块做DFT变换后，只用幅度信息重构，显示获得的图像；如果只用相位信息重构，幅度信息丢掉，显示这时获得的图像
- 熟练掌握DCT变换
 - 编程实现8*8块DCT变换,然后分别显示，当使用64个DCT系数里面的一些系数进行重构，例如，按照zigzag扫描方式，保留前k个系数，其余系数置0，重构，比较其质量；同时熟悉采用FFT来实现DCT变换的思想，并显示2维8*8DCT变换的基函数图像
- 预习图像直方图的统计和显示
- 有余力的同学
 - DCT变换的一个小应用-估计经历过JPEG压缩的BMP图像的压缩历史



1	2	3	4	5	6	7	8
9	10	11	12	13	14	15	16
17	18	19	20	21	22	23	24
25	26	27	28	29	30	31	32
33	34	35	36	37	38	39	40
41	42	43	44	45	46	47	48
49	50	51	52	53	54	55	56
57	58	59	60	61	62	63	64



- 理解为什么傅里叶变换后要通过fftshift来平移系数
 - 在二维的情况下，图像的直流分量才能显示在中心
 - 如果原始图像乘上一个 ± 1 相交替的矩阵，这时候也可以起到平移的作用？

■ Experiment 2

- Implement 8×8 DCT transform, and then each input image is transformed into transformed image in the DCT domain
- Give the basis functions of 8×8 DCT transform, and display the basis image
- Compare the time complexity of different implementation, such as, with FFT and without FFT

熟悉图像变换的思想



■ 空间

- 内积
- 范数
- 完备, 柯西序列, 收敛序列
 - 有理数空间
 - 开区间
- Euclid空间
- 度量空间
 - 完备的内积空间称为Hilbert空间
 - 完备的赋范空间称为Banach空间
 - 任何一个Hilbert空间都是Banach空间
- 拓扑空间

■ 基

■ 框架

■ 线性代数

- 一个内积空间的正交基(orthogonal basis)是元素两两正交的基,称基中的元素为基向量。如果一个正交基的基向量的模长都是单位长度1, 则称这正交基为标准正交基 (Orthonormal basis)

■ Euclid空间

- Gram-Schmidt正交化

■ 每个Hilbert空间都有基(base)

函数在空间上的展开



- 有限维向量空间
 - 任意向量均可展开为基中元素的线性组合
- 无限维空间
 - 采用正交基展开
 - 由正交基扩展出空间
- Banach空间不一定有基
 - ?
 - 充要条件

■ Frame

A family of function $(\varphi_j)_{j \in J}$ in a Hilbert space H is called a frame if there exist $A > 0, B < \infty$

so that, for all f in H ,

$$A\|f\|^2 \leq \sum_{j \in J} |\langle f, \varphi_j \rangle|^2 \leq B\|f\|^2$$

We call A and B the frame bounds.

■ Tight frame

It is a tight frame if $A = B$

$$\sum_{j \in J} |\langle f, \varphi_j \rangle|^2 = A\|f\|^2$$

It is easy to see that:

$$A \langle f, g \rangle = \sum_j \langle f, \varphi_j \rangle \langle \varphi_j, g \rangle$$

Namely

$$f = A^{-1} \sum_j \langle f, \varphi_j \rangle \varphi_j$$

An example



- Frames do not mean to an orthonormal basis

Take $H = \mathbf{C}^2$, $e_1 = (0,1)$, $e_2 = (-\frac{\sqrt{3}}{2}, -\frac{1}{2})$, $e_3 = (\frac{\sqrt{3}}{2}, -\frac{1}{2})$, for any $v = (v_1, v_2)$ in H

$$\sum_{j=1}^3 |\langle v, e_j \rangle|^2 = |v_2|^2 + \left| -\frac{\sqrt{3}}{2} v_1 - \frac{1}{2} v_2 \right|^2 + \left| \frac{\sqrt{3}}{2} v_1 - \frac{1}{2} v_2 \right|^2 = \frac{3}{2} [|v_1|^2 + |v_2|^2]$$

第三章 图像变换

Chapter 3 Image Transform

为什么会有2N?

✎ 3-7 离散余弦变换 Discrete Cosine Transform

📖 一. 单维离散余弦变换 (DCT)

正向核:

why?

$$\begin{cases} g(x, 0) = \frac{1}{\sqrt{N}} \\ g(x, u) = \sqrt{\frac{2}{N}} \cos \frac{(2x+1)u\pi}{2N}, \end{cases} \begin{matrix} x = 0, 1, 2, \dots, N-1 \\ u = 1, 2, \dots, N-1 \end{matrix}$$

则DCT,

$$\begin{cases} C(0) = \frac{1}{\sqrt{N}} \sum_{x=0}^{N-1} f(x) \\ C(u) = \sqrt{\frac{2}{N}} \sum_{x=0}^{N-1} f(x) \cos \frac{(2x+1)u\pi}{2N}, \end{cases} u = 1, 2, \dots, N-1$$

■ FFT

Each function $f(t)$ in time domain can be written as a sum of two parts, namely odd and even function ($f_o(t)$ and $f_e(t)$).

In fact,

$$f_o(t) = \frac{1}{2}(f(t) - f(-t)); f_e(t) = \frac{1}{2}(f(t) + f(-t))$$

Hence, $f(t) = f_o(t) + f_e(t)$

■ DCT

- A special case of DFT

$$\begin{aligned} F(u) &= \int_{-\infty}^{+\infty} f(t) e^{-j2\pi ut} dt \\ &= \int_{-\infty}^{+\infty} [f_e(t) + f_o(t)] e^{-j2\pi ut} dt \\ &= \int_{-\infty}^{+\infty} [f_e(t) + f_o(t)] [\cos(2\pi ut) - j \sin(2\pi ut)] dt \\ &= \int_{-\infty}^{+\infty} f_e(t) \cos(2\pi ut) dt + \int_{-\infty}^{+\infty} f_o(t) \cos(2\pi ut) dt \\ &\quad - j \int_{-\infty}^{+\infty} f_e(t) \sin(2\pi ut) dt - j \int_{-\infty}^{+\infty} f_o(t) \sin(2\pi ut) dt \\ &= \int_{-\infty}^{+\infty} f_e(t) \cos(2\pi ut) dt - j \int_{-\infty}^{+\infty} f_o(t) \sin(2\pi ut) dt \end{aligned}$$

一般情况信号不是偶的？

■ 平移情况

$x(t-t_0)$ is an even function, then

$$\begin{aligned} X_e(u) &= F[x_{t_0}(t)] = \int_{-\infty}^{+\infty} x(t-t_0) \cos(2\pi ut) dt \\ &= \int_{-\infty}^{+\infty} x(s) \cos(2\pi u(s+t_0)) ds \\ &= \int_{-\infty}^{+\infty} x(t) \cos(2\pi u(t+t_0)) dt \end{aligned}$$

Its invert transform is:

$$\begin{aligned} x(t-t_0) &= x_{t_0}(t) = F^{-1}[X_e(u)] \\ &= \int_{-\infty}^{+\infty} X_e(u) e^{j2\pi ut} du \\ &= \int_{-\infty}^{+\infty} X_e(u) \cos(2\pi ut) du \end{aligned}$$

$$\text{Hence, } x(t) = \int_{-\infty}^{+\infty} X_e(u) \cos(2\pi u(t+t_0)) du$$

■ 为什么DCT的公式中有2N

Consider one sequence $x(n) = x(nT_s)$ with length N

Symmetric expansion $x_1(n) : x(-n-1), x(n)$,

Shifting a half unit:

$$x_e(n) = x_1((n-1/2)T_s), -N \leq n \leq N-1$$

Then

$$\begin{aligned} X_e(k) &= \sum_{n=-N}^{N-1} x_e(n) \cos\left(\frac{2\pi kn}{2N}\right) = \sum_{n=-N}^{N-1} x_1(n) \cos\left(\frac{2\pi k(n+1/2)}{2N}\right) \\ &= \sum_{n=-N}^{N-1} x_1(n) \cos\left(\frac{\pi k(2n+1)}{2N}\right) \end{aligned}$$

Where $-N \leq k \leq N-1$

- 1. Since $X_e(k)$ is an even sequence and the symmetry of $x_1(n)$, $X_e(k)$ can be divided into two equivalent

parts: $\sum_{n=-N}^{-1} (\cdot)$ and $\sum_{n=0}^{N-1} (\cdot)$.

So:

■ 2

$$\begin{aligned}
 X_e(k) &= 2 \sum_{n=0}^{N-1} x_1(n) \cos\left(\frac{\pi k(2n+1)}{2N}\right) & x(n) &= \frac{2}{2N} \sum_{k=0}^{N-1} X_e(k) \cos\left(\frac{\pi k(2n+1)}{2N}\right) \\
 &= 2 \sum_{n=0}^{N-1} x(n) \cos\left(\frac{\pi k(2n+1)}{2N}\right) & &= \frac{1}{N} \sum_{n=0}^{N-1} X_e(k) \cos\left(\frac{\pi k(2n+1)}{2N}\right)
 \end{aligned}$$

Where $0 \leq k \leq N-1$

Where $0 \leq n \leq N-1$

Uniform Even DCT



$$X(k) = \alpha_N(k) \sum_{n=0}^{N-1} x(n) \cos\left(\frac{\pi k(2n+1)}{2N}\right), k = 0 \dots N-1$$

where

$$x(n) = \sum_{k=0}^{N-1} \alpha_N(k) X(k) \cos\left(\frac{\pi k(2n+1)}{2N}\right), n = 0 \dots N-1$$

$$\alpha_N(k) = \begin{cases} \sqrt{\frac{1}{N}}, & k = 0 \\ \sqrt{\frac{2}{N}} & \text{otherwise} \end{cases}$$

- Even DCT is an orthogonal transform
- It also is an approximation of K-L

EDCT can be implemented by FFT?

$$X(k) = \alpha_N(k) \cdot \operatorname{Re}(e^{-jk\pi/2N} \cdot \operatorname{FFT}_{2N}[x_{2N}(n)])$$

- Even DCT
- Odd DCT

$$x_e(n) = \begin{cases} x(n) & 0 \leq n \leq N-1 \\ x(-n) & -N+1 \leq n \leq -1 \end{cases}$$

$$X_e(k) = 2 \sum_{n=0}^{N-1} x_e(n) \cos\left(\frac{2\pi kn}{2N-1}\right), 0 \leq k \leq N-1$$

$$x_e(n) = 2 \sum_{k=0}^{N-1} X_e(k) \cos\left(\frac{2\pi kn}{2N-1}\right), 0 \leq n \leq N-1$$

$$X(k) = \beta_N(k) \sum_{n=0}^{N-1} x(n) \cos\left(\frac{2\pi kn}{2N-1}\right), k = 0 \dots N-1$$

$$x(n) = \sum_{k=0}^{N-1} \beta_N(k) X(k) \cos\left(\frac{2\pi kn}{2N-1}\right), n = 0 \dots N-1$$

$$\text{Where } \beta_N(k) = \begin{cases} \sqrt{\frac{2}{2N-1}}, & k = 0 \\ \sqrt{\frac{2}{2N-1}}, & \text{otherwise} \end{cases}$$

压缩历史估计

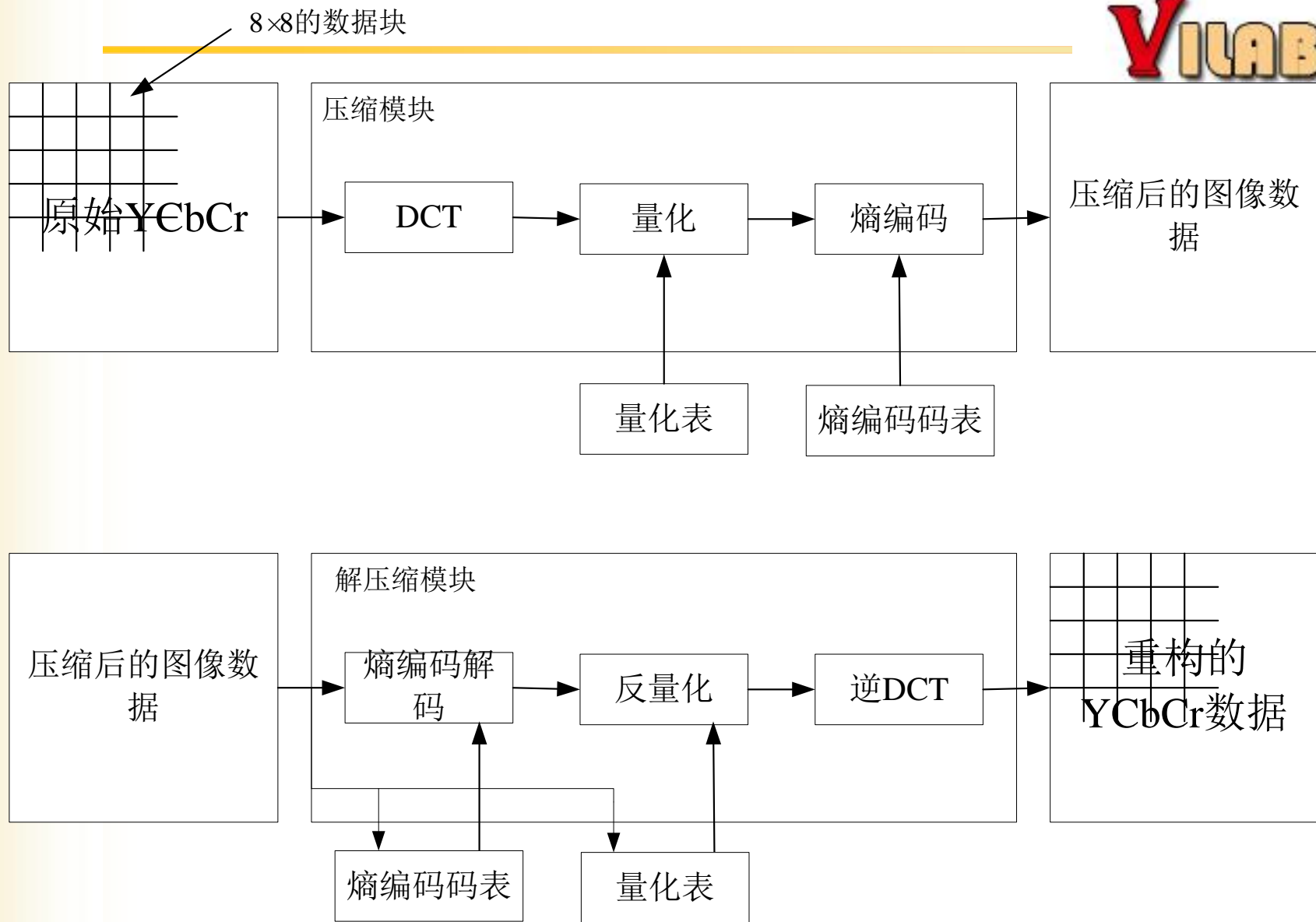


- Main idea
- JPEG compressing procedure

JPEG图像压缩的基本流程



- RGB->YCbCr
 - 注意 $Cb+128, Cr+128$
- Y-128
- 亮度和色度分量分别分块做DCT变换
- 亮度和色度分量分别用不同的量化矩阵进行量化
 - 注意量化矩阵的计算公式
- DC系数进行处理, ZigZag扫描
- 熵编码



标准量化表和量化因子



$$QuanTable = \begin{cases} round(StdQuanTable \cdot (2 - 0.02 \cdot QualityFactor)) & QualityFactor \geq 50 \\ round(StdQuanTable \cdot (50 / QualityFactor)) & QualityFactor < 50 \end{cases}$$

JPEG推荐的标
准亮度量化表

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

JPEG静态图像压缩算法

1. DCT交换:

二维分块为 $M \times N = 8 \times 8$

$$F(u, v) = \frac{1}{4} E(u) \cdot E(v) \cdot \sum_{x=0}^7 \sum_{y=0}^7 f(x, y) \left[\cos \frac{2x+1}{16} u\pi \right] \left[\cos \frac{2y+1}{16} v\pi \right]$$

有相应的快速算法FDCT。

逆交换: (已知F, 求 $f(x, y)$)

$$f(x, y) = \frac{1}{4} \left[\sum_{u=0}^7 \sum_{v=0}^7 E(u) \cdot E(v) \cdot F(u, v) \left[\cos \frac{2x+1}{16} u\pi \right] \left[\cos \frac{2y+1}{16} v\pi \right] \right]$$

有相应的快速算法FIDCT。

JPEG静态图像压缩算法

例：从某图象中取一 8×8 的块，数据为：

139	144	149	153	155	155	155	155
144	151	153	156	159	156	156	156
150	155	156	163	158	156	156	156
159	161	162	160	160	159	159	159
159	160	161	162	162	155	155	155
161	161	161	161	160	157	157	157
162	162	161	163	162	157	157	157
162	162	161	162	163	158	158	158

JPEG静态图像压缩算法



计算得到 DCT 系数为:

235.6	-1.0	-12.1	-5.2	2.1	-1.7	-2.7	1.3
-22.6	-17.5	-6.2	-3.2	-2.9	-0.1	-0.4	-1.2
-10.9	-9.3	-1.6	1.5	0.2	-0.9	-0.6	-0.1
-7.1	-1.9	0.2	1.5	0.9	-0.1	0.0	0.3
-0.6	-0.8	1.5	1.6	-0.1	-0.7	0.6	1.3
1.8	-0.2	1.6	-0.3	-0.8	1.5	1.0	-1.0
-1.3	-0.4	-0.3	-1.5	-0.5	1.7	1.1	-0.8
-2.6	-1.6	-3.8	-1.8	1.9	1.2	-0.6	-0.4

可见数据集中于左上角.

JPEG静态图像压缩算法



2.量化: DCT 系数 \rightarrow DC, 多对一的变换, $f(x,y)\rightarrow F(U,V)$, 等步长地截取数据。

$$F^Q(U,V) = \text{INT}(\text{ROUND}(F(U,V)/Q(U,V))) \quad (Q \text{ 为步长})$$

量化表:

16	11	10	16	24	40	51	61
12	12	14	19	26	58	60	55
14	13	16	24	40	57	69	56
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

数:

15	0	-1	0	0	0	0	0
-2	-1	0	0	0	0	0	0
-1	-1	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0

[illegible]

JPEG静态图像压缩算法



重建图像子块:

144	146	149	152	154	156	156	156
148	150	152	154	156	156	156	156
155	156	157	158	158	157	156	155
160	161	161	162	161	159	157	155
163	163	164	163	162	160	158	156
163	164	164	164	162	160	158	157
160	161	162	162	162	161	159	158
158	159	161	161	162	161	159	158

JPEG静态图像压缩算法



JPEG静态图像压缩算法



139	144	149	153	155	155	155	155
144	151	153	156	159	156	156	156
150	155	156	163	158	156	156	156
159	161	162	160	160	159	159	159
159	160	161	162	162	155	155	155
161	161	161	161	160	157	157	157
162	162	161	163	162	157	157	157
162	162	161	162	163	158	158	158

重建图像子块:

144	146	149	152	154	156	156	156
148	150	152	154	156	156	156	156
155	156	157	158	158	157	156	155
160	161	161	162	161	159	157	155
163	163	164	163	162	160	158	156
163	164	164	164	162	160	158	157
160	161	162	162	162	161	159	158
158	159	161	161	162	161	159	158

■ References

课件资料QQ群下载