北京航空航天大學
Beijing University of Aeronautics and Astronautics

数字图像变换
快速傅立叶变换

1768 -1830

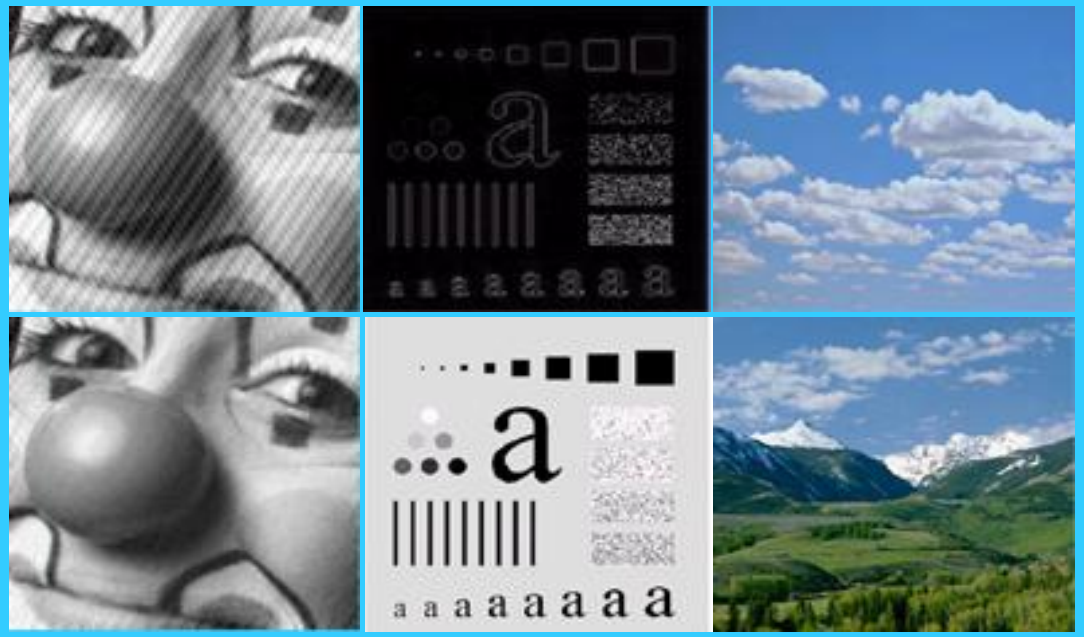
计算机学院

内容提要

- 内容回顾
- 快速傅立叶变换(FFT)
- FFT的应用

数字图像变换

- 图像变换是数字图像处理与分析中一种常用的、有效的分析手段
- 图像变换的作用
  - 简化图像处理
  - 提取图像特征
  - 增强图像信息的理解



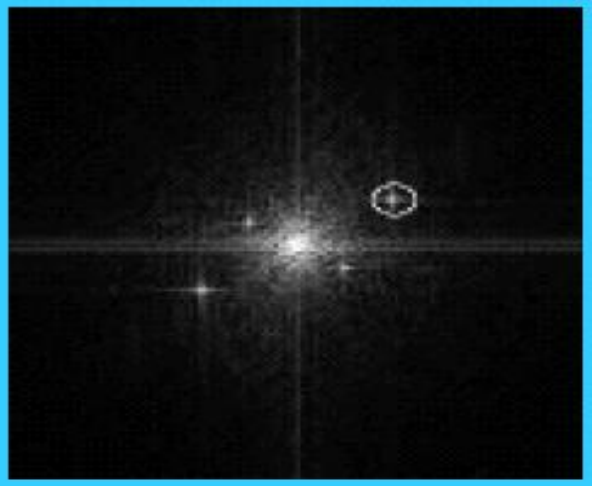
- 广泛应用在图像增强、图像恢复、特征提取、图像压缩编码和形状分析等方面

数字图像正交变换

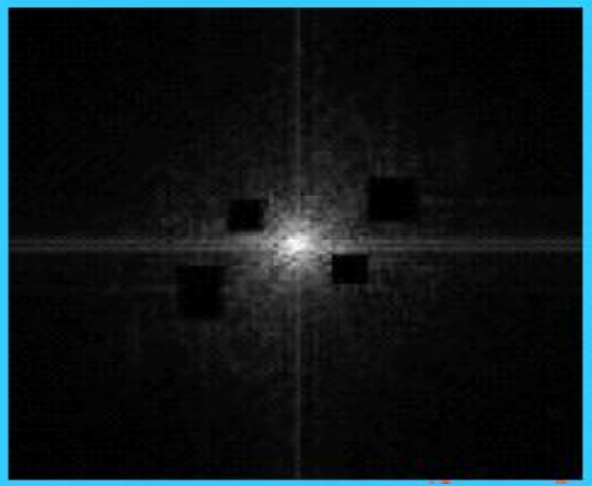
- 傅立叶变换(Fourier)
- 沃尔什变换(Walsh)
- 哈达玛变换(Hadamard)
- 离散余弦变换(DCT)
- K-L变换
- 小波变换(Wavelet)

数字图像变换

傅立叶变换



傅立叶反变换



二维傅立叶变换示例

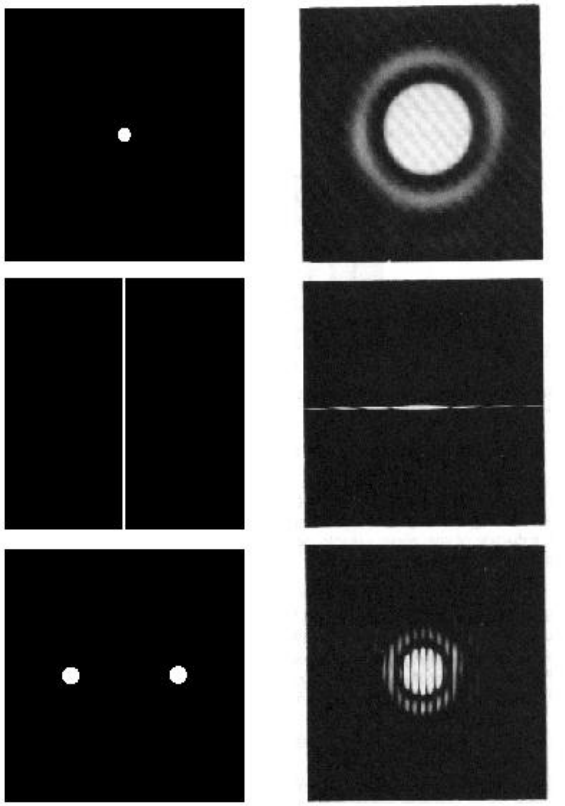


Image Domain
Frequency Domain

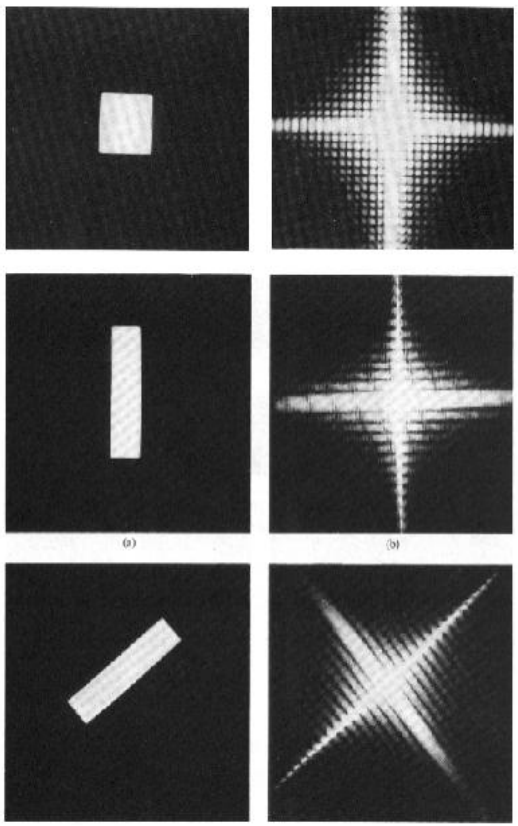


Image Domain
Frequency Domain

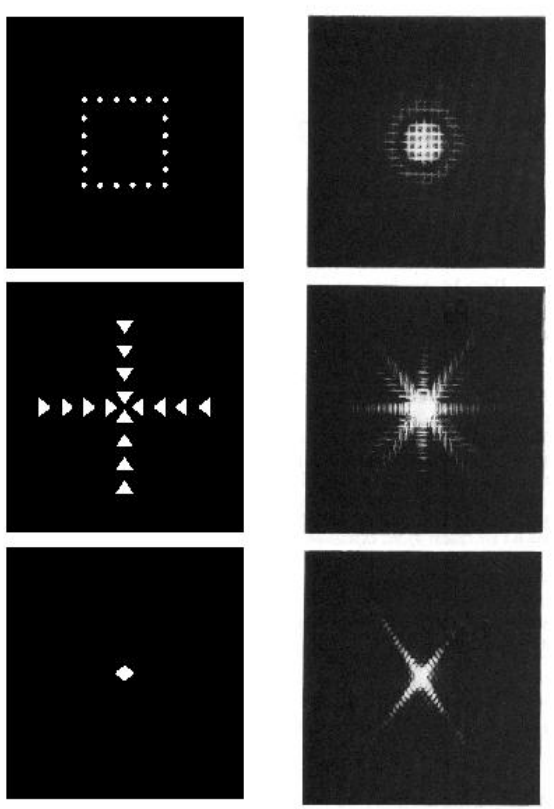
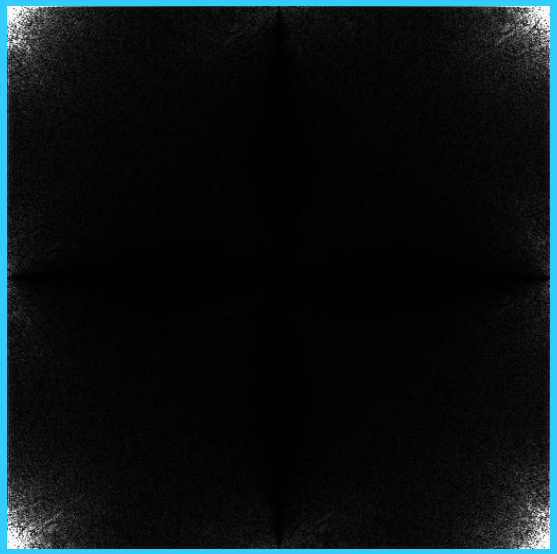
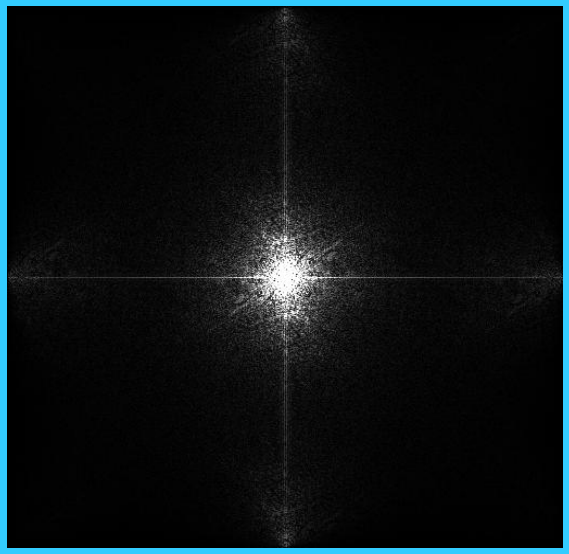


Image Domain
Frequency Domain

二维图像频域处理



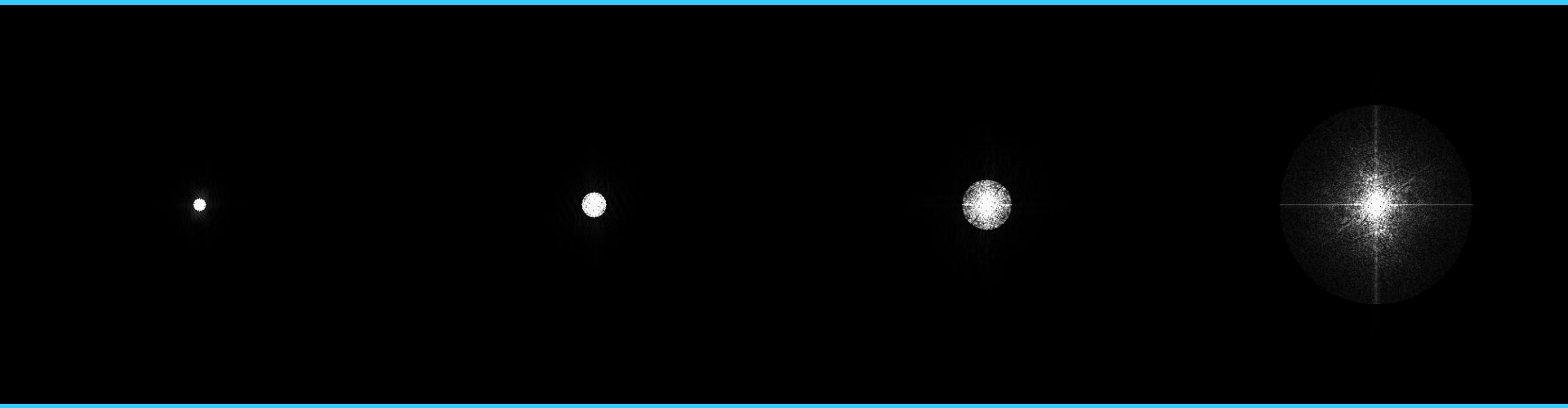
无平移的傅立叶谱



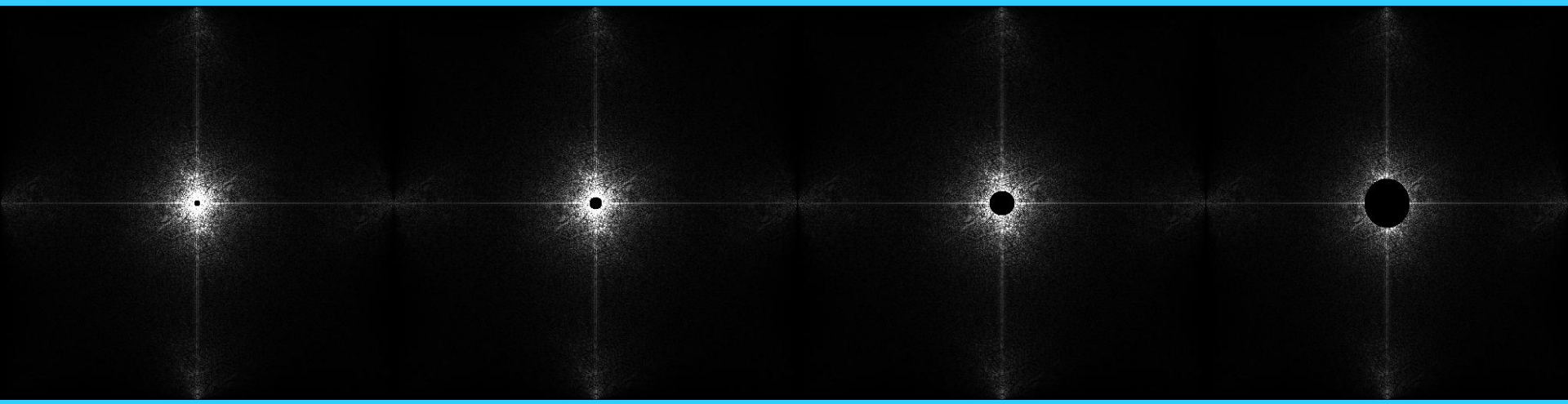
原点移到中心的傅立叶谱

平移性质表明，只要将$f(x,y)$乘以因子$(-1)^{x+y}$，再进行离散傅立叶变换，即可将图像的频谱原点$(0,0)$移动到图像中心$(M/2,N/2)$处

二维图像频域处理



二维图像频域处理



二维图像频域处理

(a) 原图　(b) 低通滤波（细微部消失）　(c) 高通滤波（突出边界）

二维傅立叶变换性质

- 可分离性
- 周期性
- 平移性
- 线性
- 共轭对称
- 相似性
- 旋转性

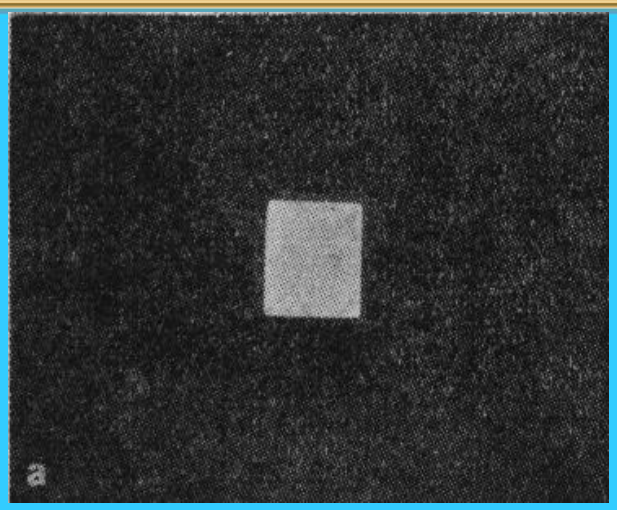
可分离性

- 二维离散傅立叶变换DFT可分离性的基本思想是：
  - 二维DFT可分离为两次一维DFT
- 应用：
  - 二维快速傅立叶算法FFT，是通过计算两次一维FFT实现的

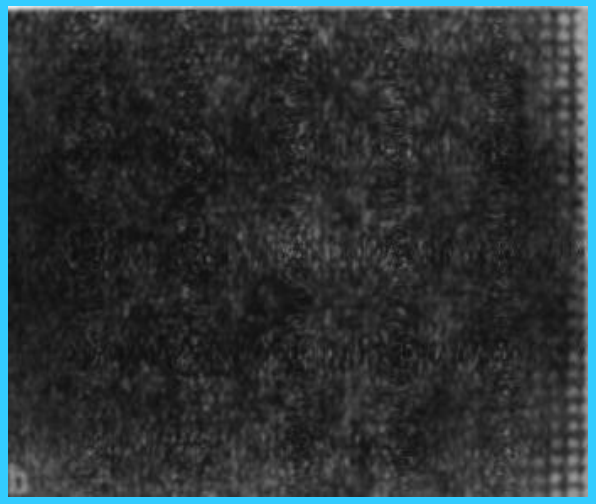
平移性-中心化

平移不改变傅立叶变换的幅值

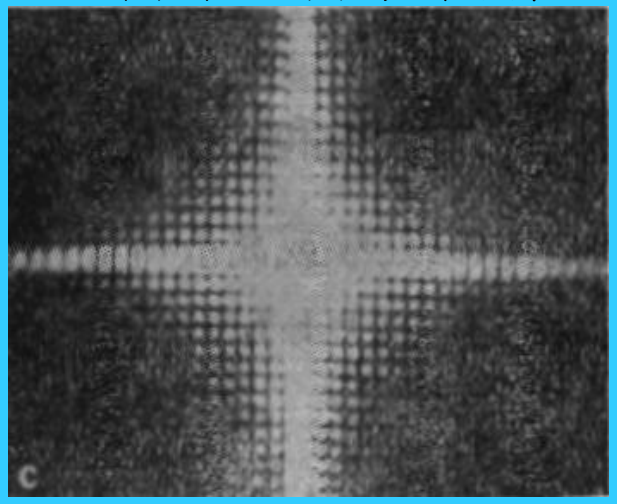
$$\Rightarrow f(x,y)\cdot(-1)^{x+y} \leftrightarrow F\left(u-\frac{M}{2},v-\frac{N}{2}\right)$$



简单的方块图像



无平移的傅里叶谱



原点移到中心的傅里叶谱

线性

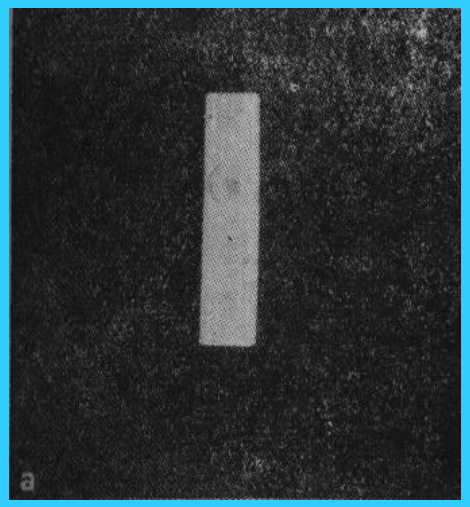
$$af_1(x, y)+bf_2(x, y) \Leftrightarrow aF_1(u, v)+bF_2(u, v)$$

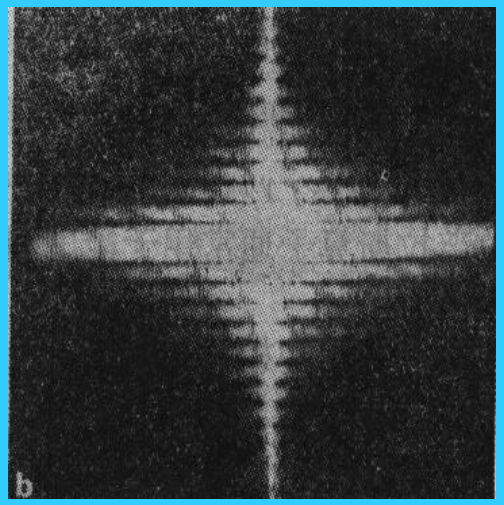
旋转不变性

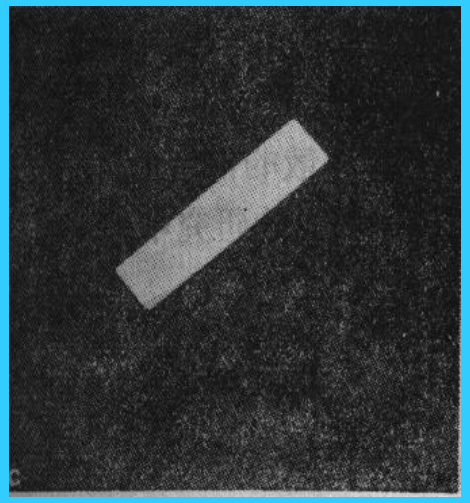
如果图像域中离散函数旋转一个角度，则在变换域中该离散傅立叶变换函数也将旋转同样角度。

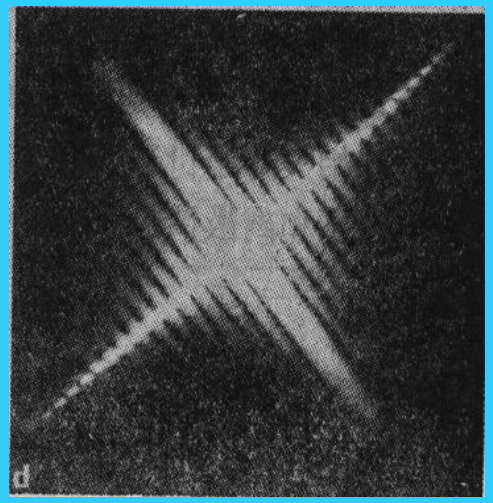
$$f(\gamma,\varphi) \Leftrightarrow F(\omega,\phi)$$

$$f(\gamma,\varphi+\alpha_0) \Leftrightarrow F(\omega,\phi+\alpha_0)$$









图像一维傅立叶变换

- 设 x(n):x(0),x(1),.....,x(N-1);
  X(m):X(0), X(1),.....,X(N-1)是数字序列,
  则序列x(n)的傅立叶变换生成序列X(m)表示如下:

正变换 $X(m)=\sum_{n=0}^{N-1}x(n)e^{-j2\pi mn/N}$

反变换 $x(n)=\frac{1}{N}\sum_{m=0}^{N-1}X(m)e^{j2\pi mn/N}$

缩写 $W=e^{\frac{-j2\pi}{N}}$ 函数W的周期为N

图像一维傅立叶变换

$$\begin{bmatrix} X(0) \\ X(1) \\ X(2) \\ X(3) \\ X(4) \\ X(5) \\ X(6) \\ X(7) \end{bmatrix} = \begin{bmatrix} W^{0*0} & W^{0*1} & W^{0*2} & W^{0*3} & W^{0*4} & W^{0*5} & W^{0*6} & W^{0*7} \\ W^{1*0} & W^{1*1} & W^{1*2} & W^{1*3} & W^{1*4} & W^{1*5} & W^{1*6} & W^{1*7} \\ W^{2*0} & W^{2*1} & W^{2*2} & W^{2*3} & W^{2*4} & W^{2*5} & W^{2*6} & W^{2*7} \\ W^{3*0} & W^{3*1} & W^{3*2} & W^{3*3} & W^{3*4} & W^{3*5} & W^{3*6} & W^{3*7} \\ W^{4*0} & W^{4*1} & W^{4*2} & W^{4*3} & W^{4*4} & W^{4*5} & W^{4*6} & W^{4*7} \\ W^{5*0} & W^{5*1} & W^{5*2} & W^{5*3} & W^{5*4} & W^{5*5} & W^{5*6} & W^{5*7} \\ W^{6*0} & W^{6*1} & W^{6*2} & W^{6*3} & W^{6*4} & W^{6*5} & W^{6*6} & W^{6*7} \\ W^{7*0} & W^{7*1} & W^{7*2} & W^{7*3} & W^{7*4} & W^{7*5} & W^{7*6} & W^{7*7} \end{bmatrix} \begin{bmatrix} x(0) \\ x(1) \\ x(2) \\ x(3) \\ x(4) \\ x(5) \\ x(6) \\ x(7) \end{bmatrix}$$

- x(n)是输入函数，X(m)是输出函数，N=8

$$X(m) = \sum_{n=0}^{n=7} x(n) e^{-j2\pi mn/8}$$

- 函数W周期为N=8

$$W = e^{\frac{-j2\pi}{8}}$$

$$W^0 = 1, \quad W^1 = (1-j)/\sqrt{2}$$
$$W^2 = -j \quad W^3 = (-1-j)/\sqrt{2}$$
$$W^4 = -1, \quad W^5 = (-1+j)/\sqrt{2}$$
$$W^6 = j \quad W^7 = (1+j)/\sqrt{2}$$

图像一维傅立叶变换

$$\begin{bmatrix} X(0) \\ X(1) \\ X(2) \\ X(3) \\ X(4) \\ X(5) \\ X(6) \\ X(7) \end{bmatrix} = \begin{bmatrix} W^{0*0} & W^{0*1} & W^{0*2} & W^{0*3} & W^{0*4} & W^{0*5} & W^{0*6} & W^{0*7} \\ W^{1*0} & W^{1*1} & W^{1*2} & W^{1*3} & W^{1*4} & W^{1*5} & W^{1*6} & W^{1*7} \\ W^{2*0} & W^{2*1} & W^{2*2} & W^{2*3} & W^{2*4} & W^{2*5} & W^{2*6} & W^{2*7} \\ W^{3*0} & W^{3*1} & W^{3*2} & W^{3*3} & W^{3*4} & W^{3*5} & W^{3*6} & W^{3*7} \\ W^{4*0} & W^{4*1} & W^{4*2} & W^{4*3} & W^{4*4} & W^{4*5} & W^{4*6} & W^{4*7} \\ W^{5*0} & W^{5*1} & W^{5*2} & W^{5*3} & W^{5*4} & W^{5*5} & W^{5*6} & W^{5*7} \\ W^{6*0} & W^{6*1} & W^{6*2} & W^{6*3} & W^{6*4} & W^{6*5} & W^{6*6} & W^{6*7} \\ W^{7*0} & W^{7*1} & W^{7*2} & W^{7*3} & W^{7*4} & W^{7*5} & W^{7*6} & W^{7*7} \end{bmatrix} \begin{bmatrix} x(0) \\ x(1) \\ x(2) \\ x(3) \\ x(4) \\ x(5) \\ x(6) \\ x(7) \end{bmatrix}$$

$$\begin{bmatrix} X(0) \\ X(1) \\ X(2) \\ X(3) \\ X(4) \\ X(5) \\ X(6) \end{bmatrix} = \begin{bmatrix} W^{0} & W^{0} & W^{0} & W^{0} & W^{0} & W^{0} & W^{0} & W^{0} \\ W^{0} & W^{1} & W^{2} & W^{3} & W^{4} & W^{5} & W^{6} & W^{7} \\ W^{0} & W^{2} & W^{4} & W^{6} & W^{0} & W^{2} & W^{4} & W^{6} \\ W^{0} & W^{3} & W^{6} & W^{1} & W^{4} & W^{7} & W^{2} & W^{5} \\ W^{0} & W^{4} & W^{0} & W^{4} & W^{0} & W^{4} & W^{0} & W^{4} \\ W^{0} & W^{5} & W^{2} & W^{7} & W^{4} & W^{1} & W^{6} & W^{3} \\ W^{0} & W^{6} & W^{4} & W^{2} & W^{0} & W^{6} & W^{4} & W^{2} \\ W^{0} & W^{7} & W^{6} & W^{5} & W^{4} & W^{3} & W^{2} & W^{1} \end{bmatrix} \begin{bmatrix} x(0) \\ x(1) \\ x(2) \\ x(3) \\ x(4) \\ x(5) \\ x(6) \\ x(7) \end{bmatrix}$$

内容提要

- 内容回顾
- 快速傅立叶变换(FFT)
- FFT的应用

问题的提出

X[0] X[1] X[2] X[3]

4点序列{2，3，3，2} DFT的计算复杂度

$$X[m]=\sum_{k=0}^{N-1}x[k]W_N^{km},\quad m=0,1,\cdots N-1$$

$$W_N^{km}=e^{-j2\pi\frac{km}{N}}$$

$$X[0]=2W_N^{0\times 0}+3W_N^{1\times 0}+3W_N^{2\times 0}+2W_N^{3\times 0}=10$$

$$X[1]=2W_N^{0\times 1}+3W_N^{1\times 1}+3W_N^{2\times 1}+2W_N^{3\times 1}=-1-j$$

$$X[2]=2W_N^{0}+3W_N^{2}+3W_N^{4}+2W_N^{6}=0$$

$$X[3]=2W_N^{0}+3W_N^{3}+3W_N^{6}+2W_N^{9}=-1+j$$

复数加法 $N(N-1)$　　复数乘法 N^2

如何提高DFT的运算效率?

DFT的计算复杂度分析

$$\begin{bmatrix} X(0) \\ X(1) \\ \cdots \\ X(N-1) \end{bmatrix} = \begin{bmatrix} W^{0\times 0} & W^{0\times 1} & \cdots & W^{0\times N-1} \\ W^{1\times 0} & W^{1\times 1} & \cdots & W^{1\times N-1} \\ \cdots & \cdots & \cdots & \cdots \\ W^{N-1\times 0} & W^{N-1\times 1} & \cdots & W^{N-1\times N-1} \end{bmatrix} \begin{bmatrix} x(0) \\ x(1) \\ \cdots \\ x(N-1) \end{bmatrix}$$

计算复杂性：N^2乘法，$N(N-1)$加法

FFT的发展过程

- 1850年，高斯给出了DFT有效算法
- 1942年，丹尼尔森证明了一个界长为N的傅立叶变换可以由两个界长为N/2的傅立叶变换表示。
- 1964年，库利-图基给出了快速傅立叶变换算法
- 将长序列DFT分解为短序列的DFT
- 利用旋转因子的周期性、对称性、可约性

FFT的基本思路

- 傅立叶变换中包含了较多的冗余计算

$$F(u)=\frac{1}{N}\sum_{x=0}^{N-1}f(x)e^{-j2\pi\frac{u}{N}x} \quad ,u=0,1,\ldots,N-1$$

可重写为：

$$X_m=\sum_{n=0}^{N-1}x_nW_N^{mn} \quad ,\text{其中} \quad W_N=e^{-j\frac{2\pi}{N}} \quad \text{称为旋转因子}$$

旋转因子 W_N^{km} 的性质

1) 周期性

$$e^{jx} = \cos x + j \sin x$$

$$W_N^{(k+N)m} = W_N^{k(m+N)} = W_N^{km}$$

2) 对称性

$$W_N^{mk+\frac{N}{2}} = -W_N^{mk} \qquad \left(W_N^{km}\right)^* = W_N^{-mk}$$

3) 可约性

$$W_N^{mk} = W_{nN}^{nmk}$$

$$W_N^{mk} = W_{N/n}^{mk/n}, \quad N/n\text{为整数}$$

解决问题的方法

将时域序列逐次分解为一组子序列，利用旋转因子的特性，由子序列的DFT来实现整个序列的DFT。

基2时间抽取(Decimation in time)FFT算法

$$x[k] \rightarrow \begin{cases} x[2r] \\ x[2r+1] \end{cases} \quad r = 0,1,\cdots \frac{N}{2} - 1$$

基2频率抽取(Decimation in frequency)FFT算法

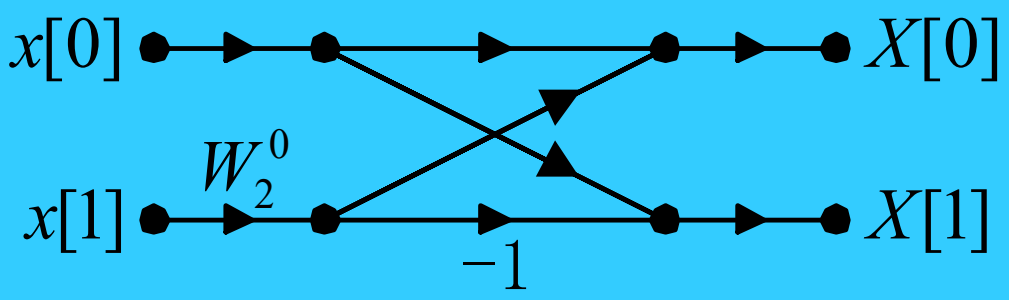
$$X[m] \rightarrow \begin{cases} X[2m] \\ X[2m+1] \end{cases}$$

基2时间抽取FFT流图

$N=2$ $\quad x[k]=\{x[0], x[1]\}$

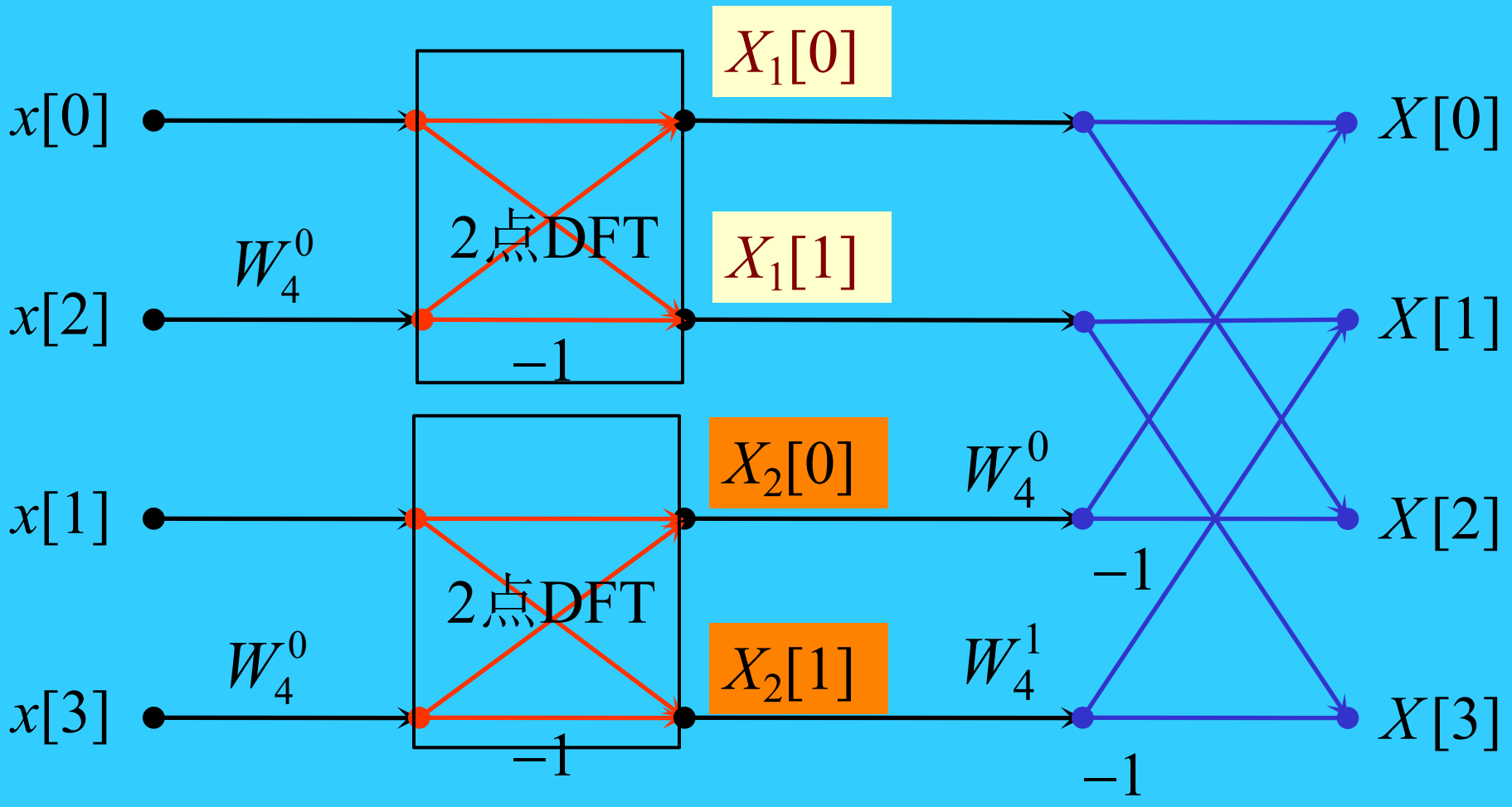
$$X[0] = x[0] + W_2^0 x[1]$$

$$X[1] = x[0] + W_2^1 x[1] = x[0] - W_2^0 x[1]$$

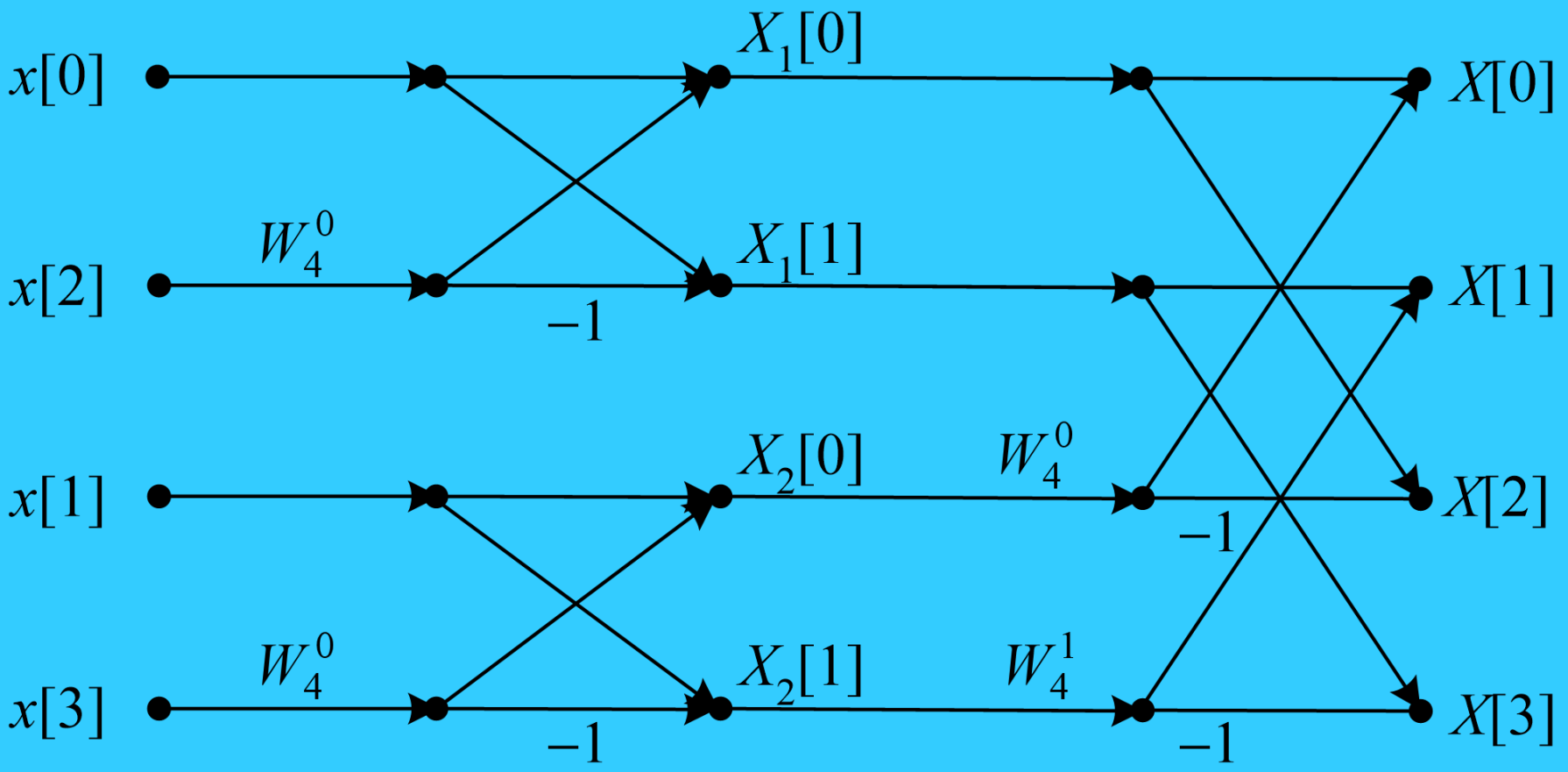


$$X[m] = X_1[m] + W_4^m X_2[m], \quad m = 0,1$$

$$X[m+2] = X_1[m] - W_4^m X_2[m], \quad m = 0,1$$

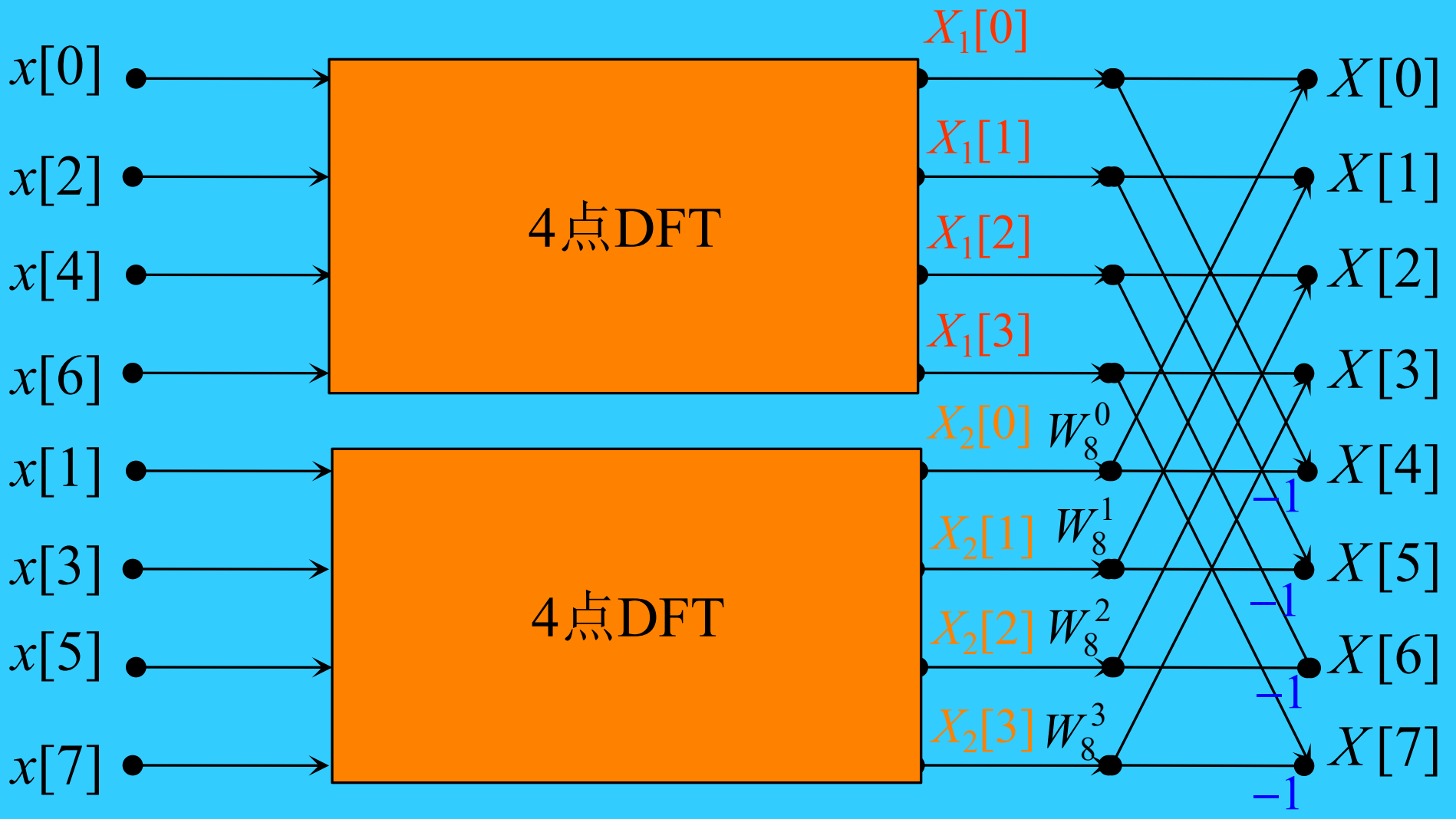


4点基2时间抽取FFT流图

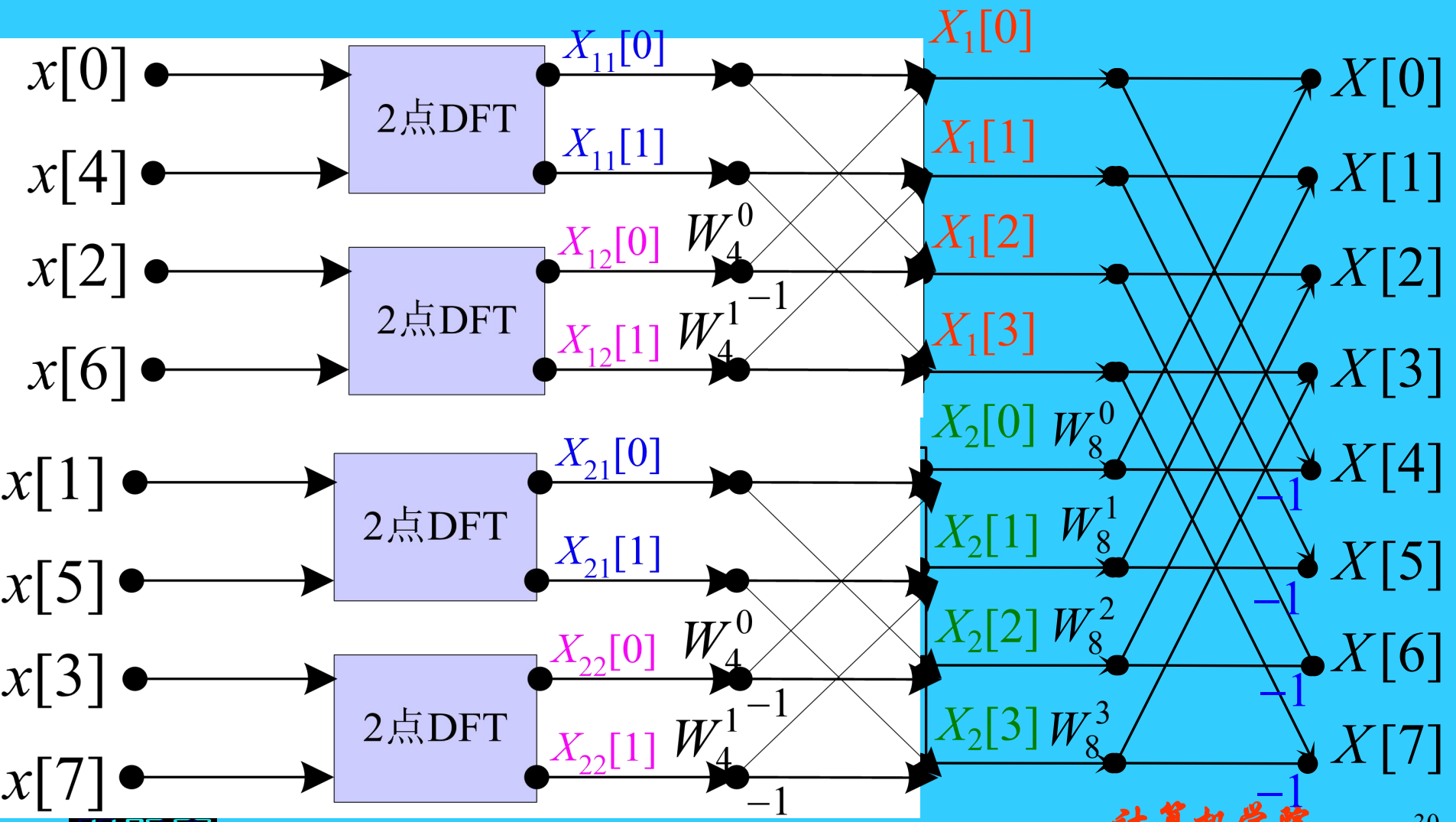


$$X[m] = X_1[m] + W_8^m X_2[m], \quad m = 0,1,2,3$$

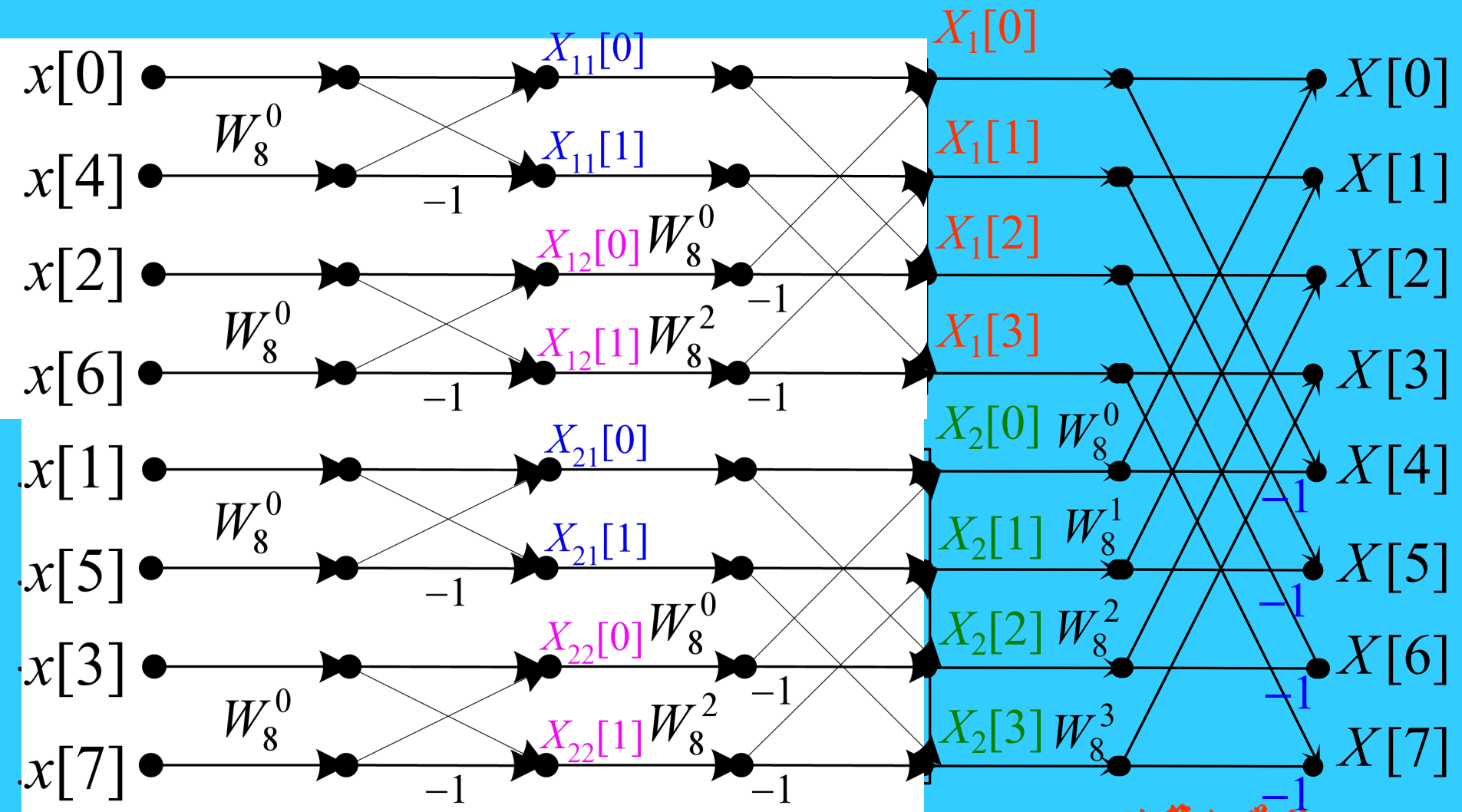
$$X[m+4] = X_1[m] - W_8^m X_2[m], \quad m = 0,1,2,3$$



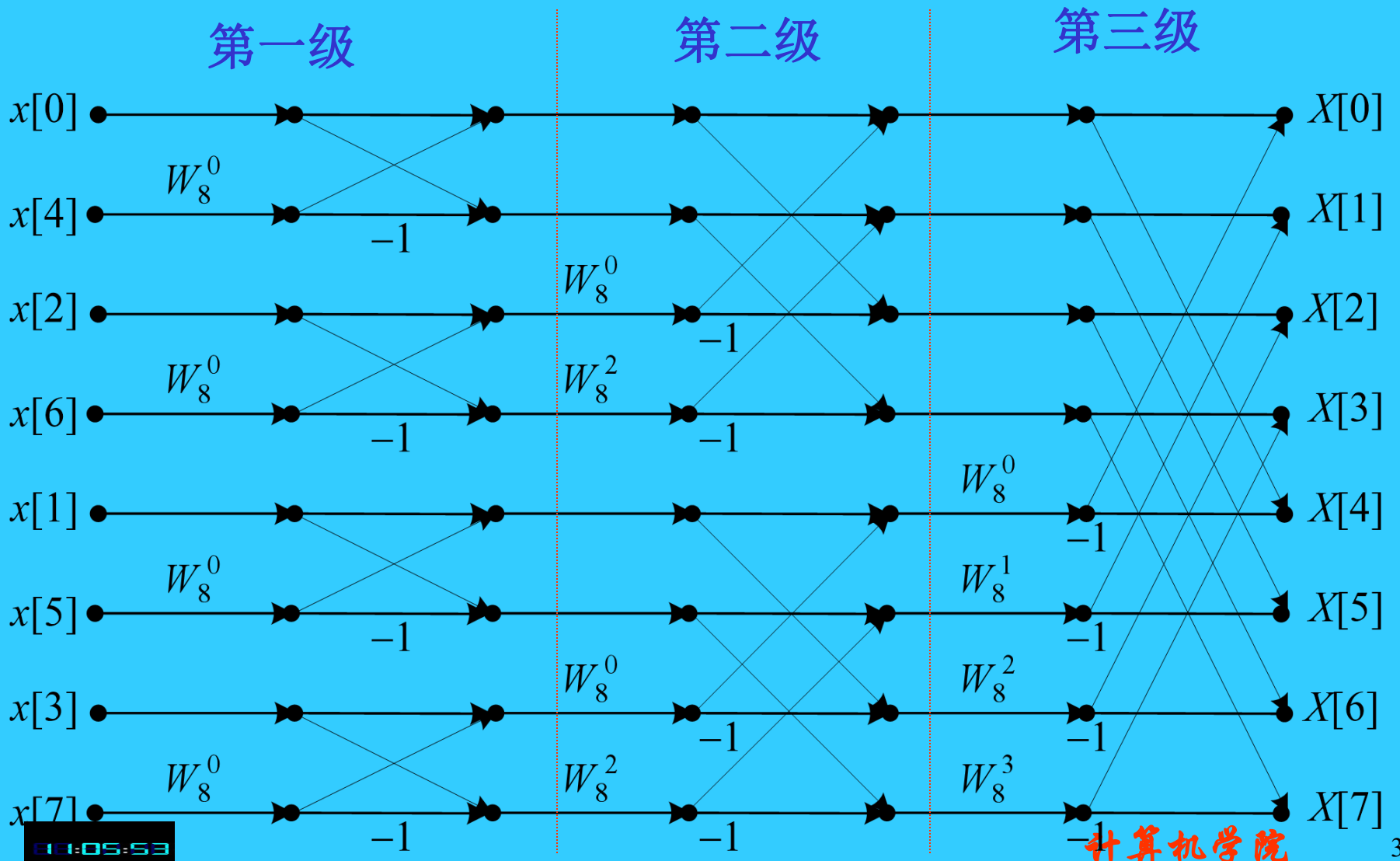
8点基2时间抽取FFT流图



8点基2时间抽取FFT流图



基2时间抽取FFT算法



FFT的公式推导

- 由傅立叶变换定义

$$X(m)=\sum_{n=0}^{N-1}x(n)W_N^{mn}$$

$$=\sum_{n=0}^{\frac{N-1}{2}}x(2n)W_N^{m(2n)}+\sum_{n=0}^{\frac{N-1}{2}}x(2n+1)W_N^{m(2n+1)}$$

$$=\sum_{n=0}^{\frac{N-1}{2}}x(2n)W_N^{m(2n)}+W_N^m\sum_{n=0}^{\frac{N-1}{2}}x(2n+1)W_N^{m(2n)}$$

FFT的公式推导

- 因为

$$W_N^{2k} = e^{(\frac{-j2\pi}{N})2k} = e^{(\frac{-j2\pi}{\frac{N}{2}})k} = W_{\frac{N}{2}}^{k}$$

- 因为周期性，所以对于所有的m，有

$$X(m) = \sum_{n=0}^{\frac{N-1}{2}} x(2n) W_N^{m(2n)} + W_N^m \sum_{n=0}^{\frac{N-1}{2}} x(2n+1) W_N^{m(2n)}$$

$$X(m) = \sum_{n=0}^{\frac{N-1}{2}} x_1(n) W_{\frac{N}{2}}^{mn} + W_N^m \sum_{n=0}^{\frac{N-1}{2}} x_2(n) W_{\frac{N}{2}}^{mn}$$

FFT的计算复杂度改进

利用FFT算法之后，任何一个N为2的整数幂（即$N= 2^r$）的DFT，都可以通过r次分解，最后成为2点的DFT来计算。r次分解构成了从x(n)到X(k)的r级迭代计算，每级由N/2个蝶形运算组成。完成一个蝶形计算需一次复数乘法和两次复数加法。

FFT的计算复杂度改进

- 乘法

$$\left(\frac{N}{2}\right)^2+\left(\frac{N}{2}\right)^2+N=2\left(\frac{N}{2}\right)^2+N$$

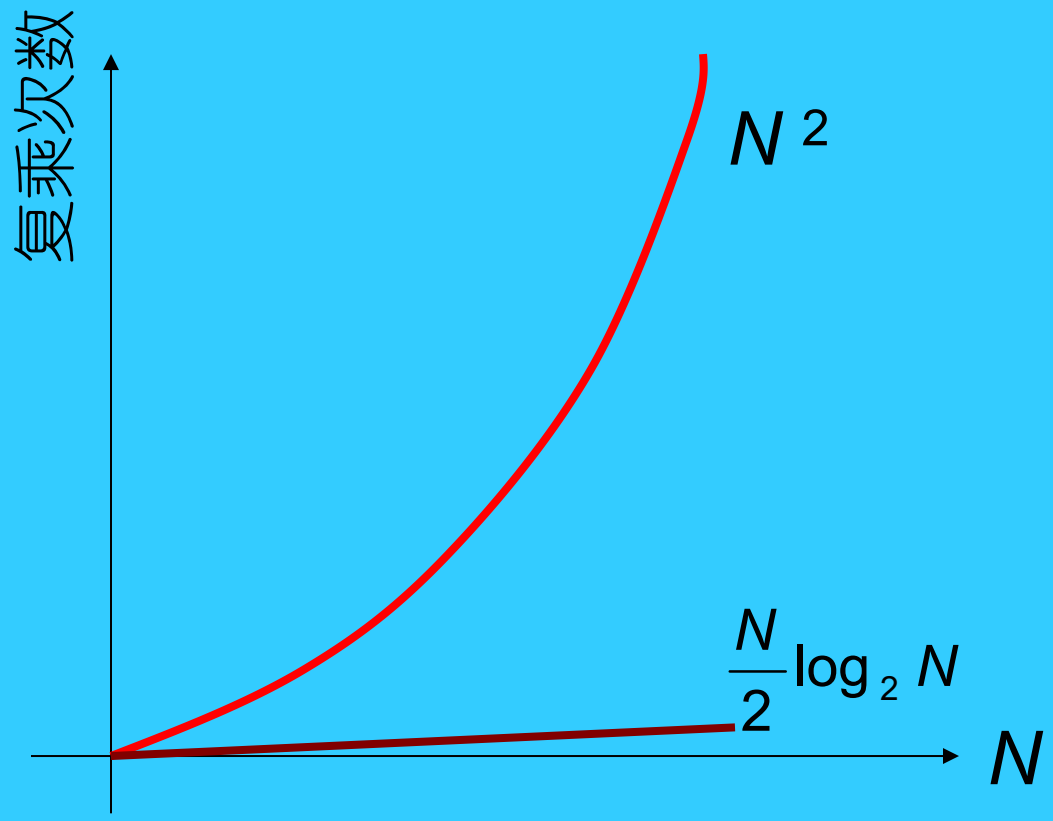
  - 经过一系列的分解2,4,8….N/2
  - 乘法计算复杂性：$N/2*\log_2 N$

FFT与DFT计算量比较

N	N*N	NlogN	（N*N）/（NlogN）
2	4	2	2.0
4	16	8	2.0
8	64	24	2.7
16	256	64	4.0
32	1024	160	6.4
64	4096	394	10.7
128	16384	896	18.3
256	65536	2048	32.0
512	262144	4608	56.9
1024	1048576	10240	102.4
2048	4194304	22528	186.2

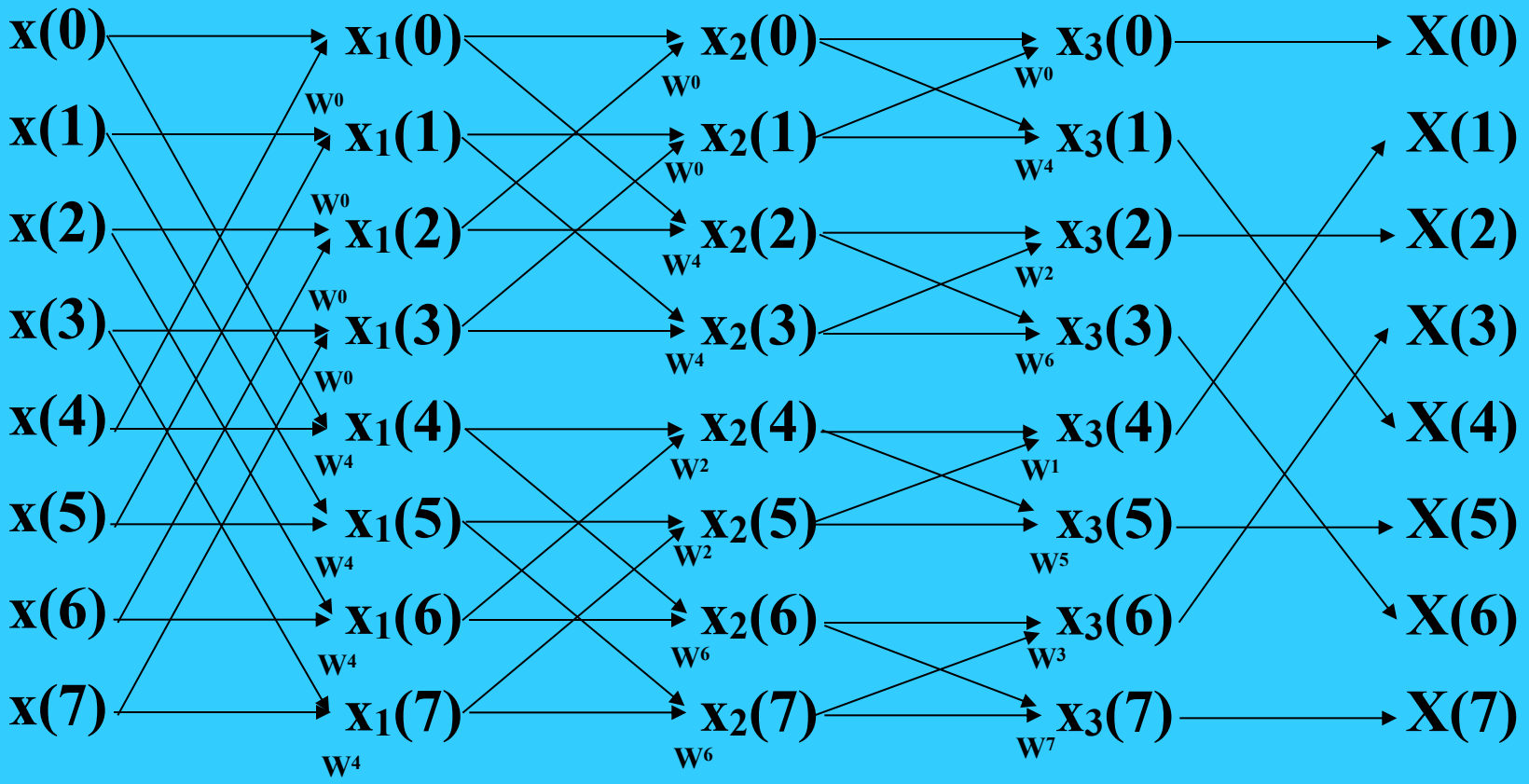
算法的计算复杂度

复乘次数 $\frac{N}{2}\log_2 N$



FFT的蝶式算法

- FFT蝶式流程图算法



FFT碟式流图算法步骤

- 迭代次数r确定
- 对偶节点的计算
- 加权系数 W_N^p 的计算
- 重新排序

FFT碟式流程图算法(1)

- 迭代次数r确定

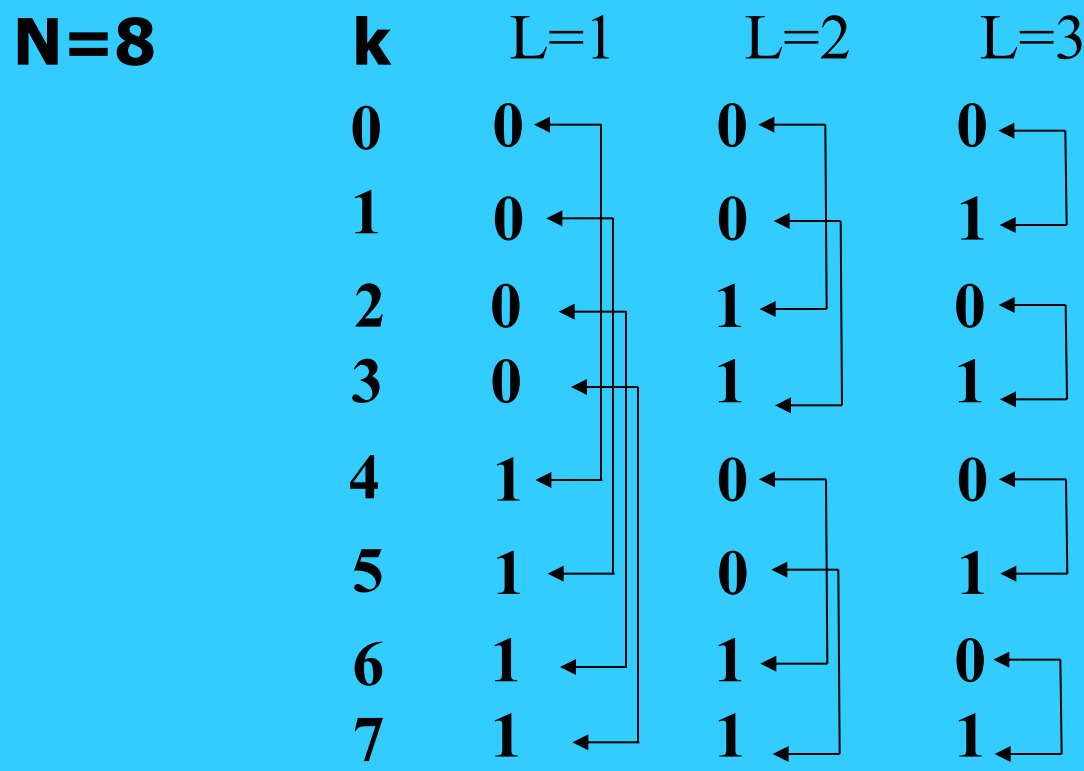
$r = \log_2 N$

FFT碟式流程图算法(2)

- 对偶节点的计算
- 节点$x_L(k)$
- L是迭代次数，k是流程图的行数
- $x_L(k)$ 的对偶节点是 $x_L(k+\frac{N}{2^L})$
- 其中$1 \leqslant L \leqslant r$，$0 \leqslant k < (N/2)$

FFT碟式流程图算法(2)

$x_L(k)$ 的对偶节点是 $x_L(k+\frac{N}{2^L})$



FFT碟式流程图算法(3)

- 加权系数 W_N^p 的计算：确定p

  (1)把k值写成r位二进制数

  (2)把这个二进制数右移r-L位

  (3)二进制数按比特位倒转

  (4)倒转后的二进制数变为十进制数

FFT碟式流程图算法(3)

- 求k=2，L=2，N=8的加权系数 W_8^p 的计算：确定p

  (1) 把k值写成r位二进制数：010

  (2) 把这个二进制数右移r-L=1位：001

  (3) 二进制数按比特位倒转：100

  (4) 倒转后的二进制数变为十进制数：4

FFT碟式流程图算法(3)

- 如果节点$x_L(k)$的加权系数是W_N^p，则$x_L(k)$的对偶节点的加权系数为$W_N^{p+\frac{N}{2}}$。
- $x_L(k) = x_{L-1}(k) + W_N^p x_{L-1}(k+\frac{N}{2^L})$

$$x_l(k+\frac{N}{2^l}) = x_{l-1}(k) + W_N^{p+\frac{N}{2}} x_{l-1}(k+\frac{N}{2^l})$$

$$x_l(k+\frac{N}{2^l}) = x_{l-1}(k) - W_N^p x_{l-1}(k+\frac{N}{2^l})$$

FFT碟式流程图算法(4)

- 重新排序
  - 将节点$x_L(k)$的k变为二进制数
    - $x_L(k)=x_L(k_{r-1}k_{r-2}\ldots k_0)$
  - 将二进制数按比特位倒转
    - $x_L(k)=x_L(k_0k_1\ldots k_{r-1})$
  - 将二进制数变为十进制数，即为重新排序位置

FFT碟式流程图算法(4)

- $x_3(0) \rightarrow x_3(000) \rightarrow x_3(000) \rightarrow x_3(0)$
- $x_3(1) \rightarrow x_3(001) \rightarrow x_3(100) \rightarrow x_3(4)$
- $x_3(2) \rightarrow x_3(010) \rightarrow x_3(010) \rightarrow x_3(2)$
- $x_3(3) \rightarrow x_3(011) \rightarrow x_3(110) \rightarrow x_3(6)$
- $x_3(4) \rightarrow x_3(100) \rightarrow x_3(001) \rightarrow x_3(1)$
- $x_3(5) \rightarrow x_3(101) \rightarrow x_3(101) \rightarrow x_3(5)$
- $x_3(6) \rightarrow x_3(110) \rightarrow x_3(011) \rightarrow x_3(3)$
- $x_3(7) \rightarrow x_3(111) \rightarrow x_3(111) \rightarrow x_3(7)$

利用FFT实现IFFT

- 利用前向变换算法进行傅立叶反变换

  在反向变换表达式两边同取共轭，并除M

$$f(x)=\sum_{u=0}^{M-1}F(u)e^{j2\pi\frac{u}{M}x}$$

$$\frac{1}{M}f^*(x)=\frac{1}{M}\sum_{u=0}^{M-1}F^*(u)e^{-j2\pi\frac{u}{M}x}$$

$$F(u)=\frac{1}{M}\sum_{x=0}^{M-1}f(x)e^{-j2\pi\frac{u}{M}x}$$

利用FFT实现IFFT

$$X[m] = DFT\{x[k]\} = \sum_{k=0}^{N-1} x[k]W_N^{mk}$$

$$x[k] = IDFT\{X[m]\} = \frac{1}{N}\sum_{m=0}^{N-1} X[m]W_N^{-mk}$$

$$x[k] = \frac{1}{N}\left(\sum_{m=0}^{N-1} X^*[m]W_N^{mk}\right)^*$$

步骤：A) 将$X[m]$取共轭

B) 用FFT流图计算$DFT\{X^*[m]\}$

C) 对B)中结果取共轭并除以N

大作业内容

- 实现图像的FFT变换和显示
- 实现FFT反变换

- 斯坦福大学公开课:傅里叶变换及其应用
  http://open.163.com/movie/2008/2/A/L/M7Q4BLENR_M7QDNTVAL.html

内容提要

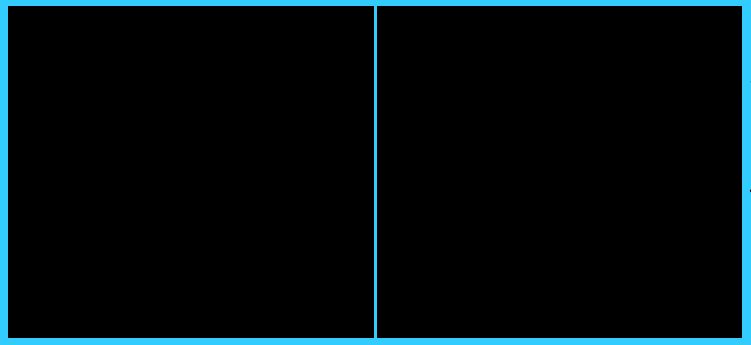
- 内容回顾
- 快速傅立叶变换(FFT)
- FFT的应用

图像傅里叶变换的物理意义

- 傅里叶变换在实际中有非常明显的物理意义，设f是一个能量有限的模拟信号，则其傅里叶变换就表示f的频谱。
- 从纯粹的数学意义上看，傅里叶变换是将一个函数转换为一系列周期函数来处理的。
- 从物理效果看，傅里叶变换是将图像从空间域转换到频率域，其逆变换是将图像从频率域转换到空间域。换句话说，傅里叶变换的物理意义是将图像的灰度分布函数变换为图像的频率分布函数。

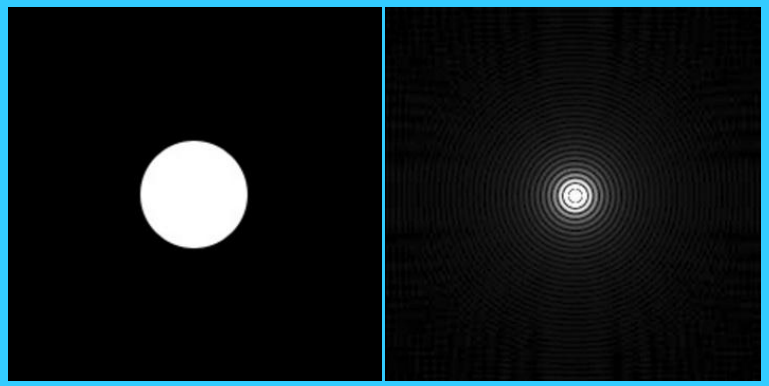
图像傅里叶变换的物理意义

- **图像的频率：**表征图像中灰度变化剧烈程度的指标，对应灰度在平面空间上的梯度。
- 如：大面积的沙漠在图像中是一片灰度变化缓慢的区域，对应的频率值很低；而对于地表属性变换剧烈的边缘区域在图像中是一片灰度变化剧烈的区域，对应的频率值较高。



全黑图——频谱图也全黑

灰色图——频谱中央单个的图像有个素色的小正方形



在图中画一个圆——频谱图呈同心圆状，趋势是越靠近中央值的绝对值越大。

图像傅里叶变换的物理意义

- 傅里叶频谱图上我们看到的明暗不一的亮点，其意义是指图像上某一点与邻域点差异的强弱，即梯度的大小，也即该点的频率的大小（可以这么理解，图像中的低频部分指低梯度的点，高频部分相反）。
- 一般来讲，梯度大则该点的亮度强，否则该点亮度弱。这样通过观察傅里叶变换后的频谱图，也叫功率图，我们就可以直观地看出图像的能量分布：如果频谱图中暗的点数更多，那么实际图像是比较柔和的（因为各点与邻域差异都不大，梯度相对较小）；反之，如果频谱图中亮的点数多，那么实际图像一定是尖锐的、边界分明且边界两边像素差异较大的。

图像傅里叶变换的物理意义

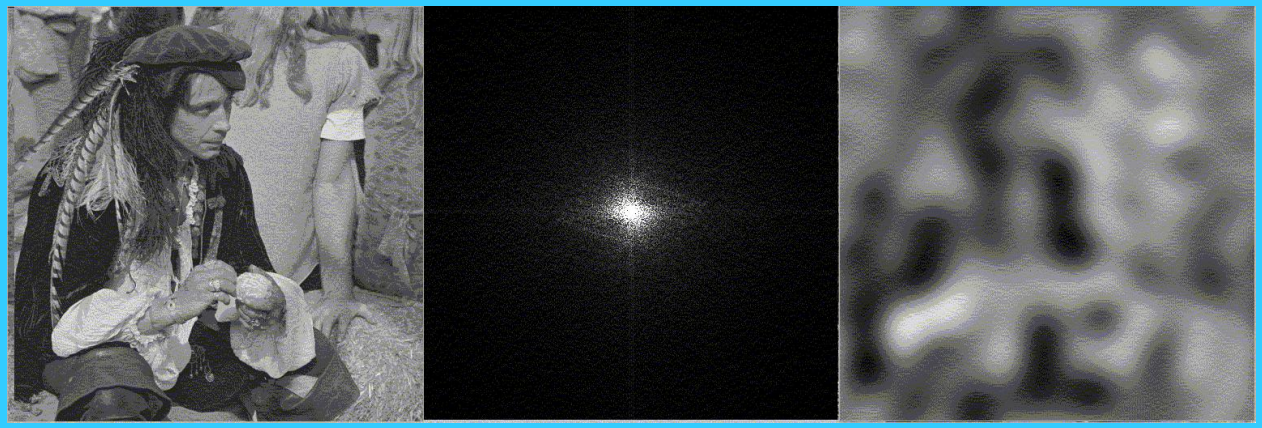
- 对频谱移频到原点以后，可以看出图像的频率分布是以原点为圆心，对称分布的。
- 将频谱移频到圆心除了可以清晰地看出图像频率分布以外，还有一个好处，它可以分离出有周期性规律的干扰信号，比如正弦干扰。
- 一幅频谱图如果带有正弦干扰，移频到原点上就可以看出，除了中心以外还存在以另一点为中心、对称分布的亮点集合，这个集合就是干扰噪音产生的。这时可以很直观的通过在该位置放置带阻滤波器消除干扰。

图像傅里叶变换的应用

- **图像增强与图像去噪**
  绝大部分噪音都是图像的高频分量，通过低通滤波器来滤除高频——噪声；边缘也是图像的高频分量，可以通过添加高频分量来增强原始图像的边缘
- **图像分割之边缘检测**
  提取图像高频分量
- **图像特征提取**
  形状特征：傅里叶描述子
  纹理特征：直接通过傅里叶系数来计算纹理特征
  其他特征：将提取的特征值进行傅里叶变换来使特征具有平移、伸缩、旋转不变性
- **图像压缩**
  可以直接通过傅里叶系数来压缩数据；常用的离散余弦变换是傅立叶变换的实变换

图像傅里叶变换的应用—频域平滑

- 图像的平滑除了在空间域中进行外，也可以在频率域中进行
- 傅立叶变换



频谱分布：

噪声→高频，图像边缘→次高频，图像→低频

图像傅里叶变换的应用—频域平滑

- 由于噪声主要集中在高频部分，为去除噪声改善图像质量，采用低通滤波器来抑制高频成分，通过低频成分，然后再进行傅立叶反变换获得滤波图像，达到平滑图像的目的
- 图像平滑数学公式：
  - G(u,v)= H(u,v)F(u,v)
  - F(u,v)原始图像傅立叶谱，H(u,v)滤波器的转移函数(频谱响应），G(u,v)平滑图像傅立叶谱
  - H(u,v)抑制高频，放过低频

图像傅里叶变换的应用—频域平滑

如果$f(x,y) \Longleftrightarrow F(u,v)$，$g(x,y) \Longleftrightarrow G(u,v)$

则$f(x,y)*g(x,y) \Longleftrightarrow F(u,v)G(x,y)$ （空域）

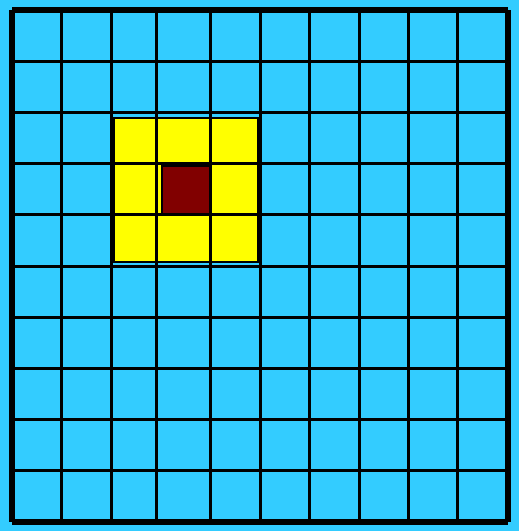
$f(x,y)g(x,y) \Longleftrightarrow F(u,v)*G(x,y)$ （频域）

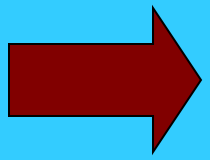
许多图像变换是卷积运算

在频域的乘积运算比在空域的卷积运算快，特别是有了快速傅里叶变换后，效果更加明显

模板运算与卷积定理

在时域内做模板运算，实际上就是对图像进行卷积。模板运算是图像处理一个很重要的处理过程，很多图像处理过程，比如增强/去噪、边缘检测中普遍用到。



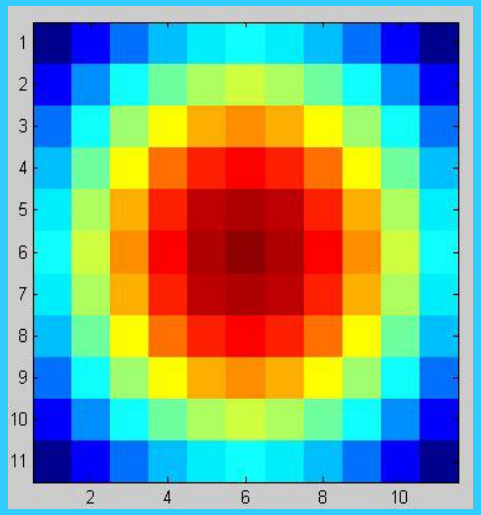


(m-1,n-1)	(m-1,n)	(m-1,n+1)
(m,n-1)	(m,n)	(m,n+1)
(m+1,n-1)	(m+1,n)	(m+1,n+1)

图像傅里叶变换的应用—频域平滑

$$\frac{1}{16} \times \begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix}$$

高斯滤波



- 离散一维卷积

对于两个长度为m和n的序列$f(i)$和$g(j)$,

$$h(i)=f(i)*g(i)=\sum_j f(j)g(i-j)$$

给出长度为$N=m+n-1$的输出序列。

其矩阵计算形式为

$$\mathbf{h}=\mathbf{g}\bullet\mathbf{f}=\begin{bmatrix} g_p(1) & g_p(N) & \cdots & g_p(2) \\ g_p(2) & g_p(1) & \cdots & g_p(3) \\ \vdots & \vdots & \vdots & \vdots \\ g_p(N) & g_p(N-1) & \cdots & g_p(1) \end{bmatrix}\begin{bmatrix} f_p(1) \\ f_p(2) \\ \vdots \\ f_p(N) \end{bmatrix}$$

卷积

- 二维卷积和离散二维卷积
  - 二维卷积定义

$$h(x,y)=f*g=\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}f(u,v)g(x-u,y-v)dudv$$

  - 离散二维卷积定义

$$H=F*G$$

$$H(i,j)=\sum_{m}\sum_{n}F(m,n)G(i-m,j-n)$$

模板运算与卷积定理

根据卷积定理，时域卷积等价与频域乘积。因此，在时域内对图像做模板运算就等效于在频域内对图像做滤波处理。

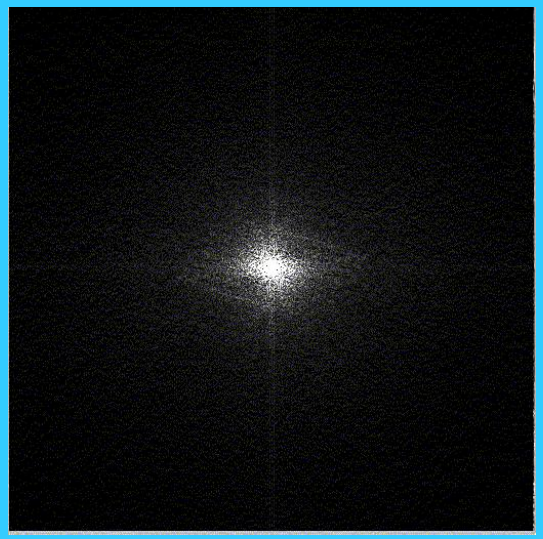
比如一个均值模板，其频域响应为一个低通滤波器；在时域内对图像作均值滤波就等效于在频域内对图像用均值模板的频域响应对图像的频域响应作一个低通滤波。

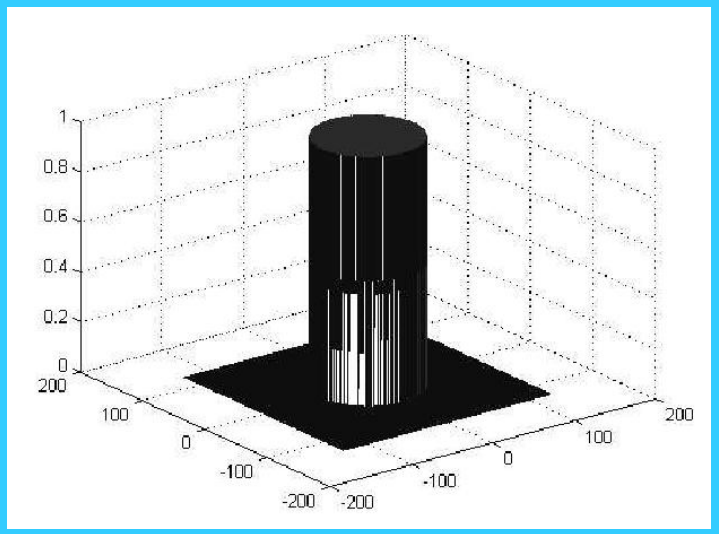
理想低通滤波器

- 理想低通滤波器的转移函数：

$$H(u,v)=\begin{cases}1 & D(u,v)<D_0(u,v)\\ 0 & D(u,v)\geq D_0(u,v)\end{cases}$$

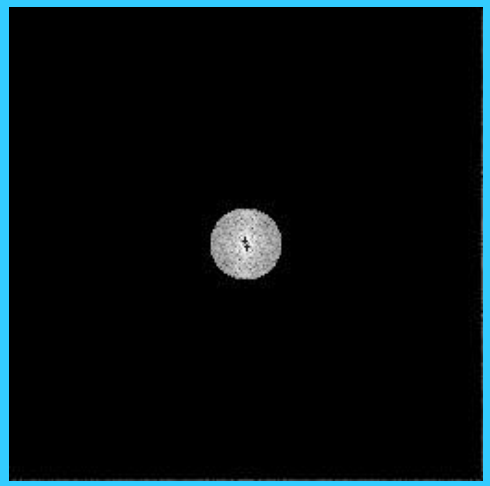
- D_0是截止频率，$D(u,v)=(u^2+v^2)^{1/2}$是点(u,v)到原点距离



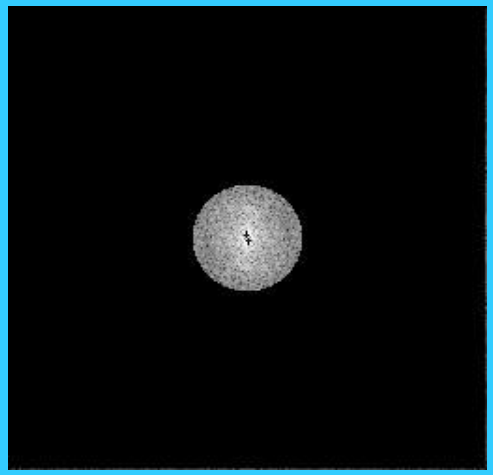


理想低通滤波器

原图像



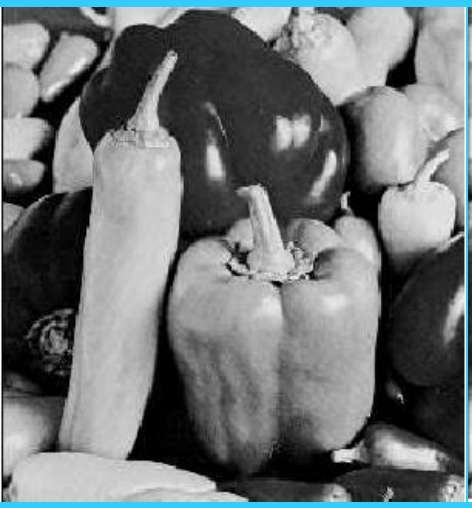
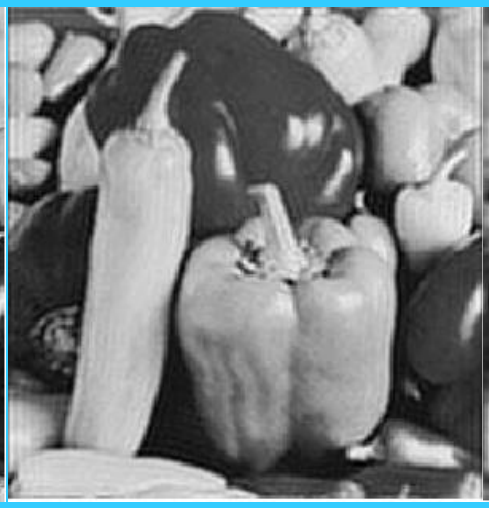
20



30

理想低通滤波器

理想低通滤波器

r_0=57　　r_0=36　　r_0=26

图像傅里叶变换的应用—图像特征提取

- 图像目标识别：首先需要抽取目标的特征，然后用适当的数学表示对目标进行描述。
- 对目标特征提取的算子称为目标检测子，对目标描述的算子称为描述子。
- 傅里叶描述子

傅里叶描述子简介

- 图像的目标区域的边界是一条封闭的曲线，因此相对于边界上某一固定的起始点来说，沿边界曲线上的一个动点的坐标变化则是一个周期函数。
- 通过规范化之后，这个周期函数可以展开成傅里叶级数．而傅里叶级数中的一系列系数是直接与边界曲线的形状有关的，可作为形状的描述，称为傅里叶描绘子。
- 目标区域边界的象素点可以用以弧长为函数的曲线切线角来表示，也可以用复变函数来表示。

傅里叶描述子定义

- 假设C是复平面上的封闭曲线（边界）。以逆时针方向沿着这个曲线保持恒定的速度移动，得到一个复函数z(t)，这里t是时间变量。
- 速度应该选择为使得环绕边界一周的时间为2π；然后沿曲线做多次里边得到一个周期为2π的周期函数。这就允许了z(t)的傅里叶表示：

$$z(t) = \sum_n T_n e^{\mathrm{int}}$$

- 其中，级数 T_n 称为曲线C的傅里叶描述子。

傅里叶描述子概念

- 考虑到曲线距离s对照于时间会更有用，因此做如下变换：

$$t = 2\pi s / L$$

- 其中L是曲线长度。傅里叶描述子 T_n 则表示如下：

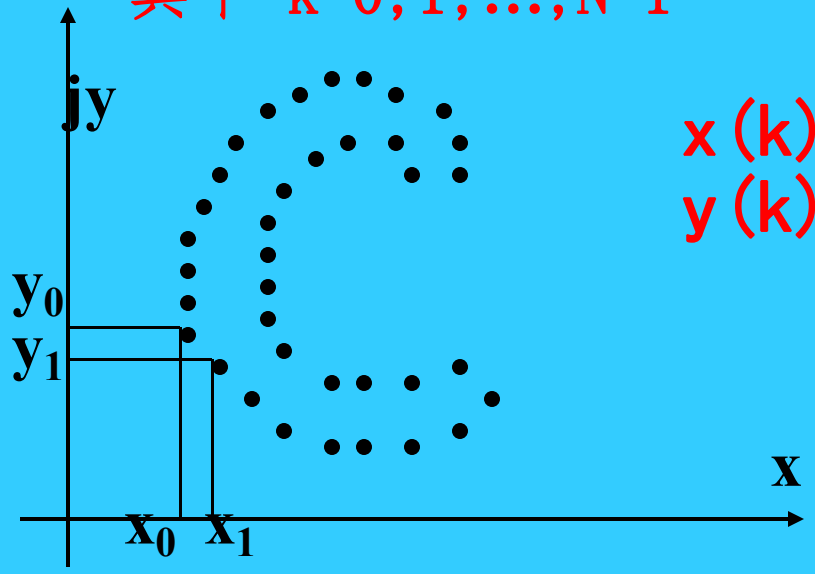
$$T_n = \frac{1}{L}\int_0^L z(s)e^{-i(2\pi/L)ns}ds$$

- 对傅里叶描述子 T_n 进行傅里叶反变换可重构回原轮廓曲线
- 傅里叶描述子反映原曲线的形状特征

- 傅立叶描述子
  - 1）基本思想：

（1）对于XY平面上的每个边界点，将其坐标用复数表示为：　　$s(k) = x(k) + jy(k)$

其中 $k=0,1,\ldots,N-1$



$$x(k) = x_k$$
$$y(k) = y_k$$

- 傅立叶描述子
  - 1）基本思想：

    （2）进行离散傅立叶变换

$$a(u) = 1/N \sum_{k=0}^{N-1} s(k)\exp(-j2\pi uk/N) \quad u=0,1,\ldots,N-1$$

$$s(k) = \sum_{u=0}^{N-1} a(u)\exp(j2\pi uk/N) \quad k=0,1,\ldots,N-1$$

a(u)被称为边界的傅立叶描述子

- 傅立叶描述子
  - 1）基本思想：

（3）选取整数 M≤N-1，进行逆傅立叶变换（重构）

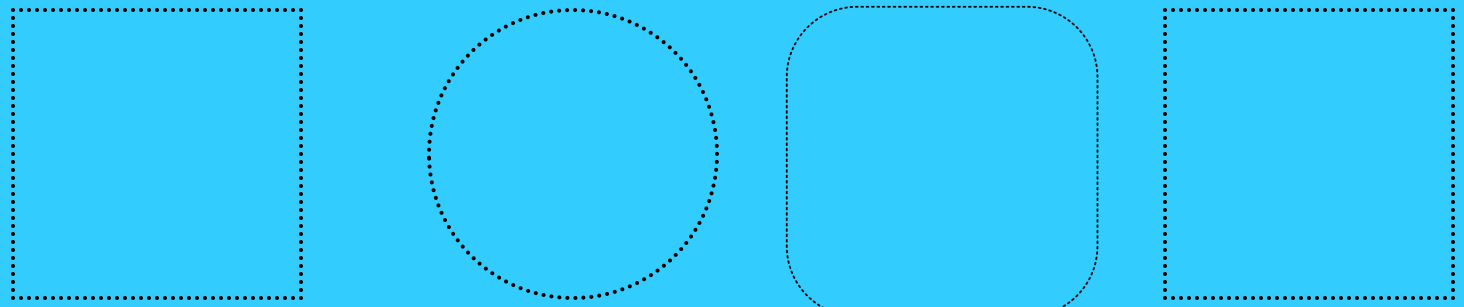
$$s'(k) = \sum_{u=0}^{M-1} a(u)\exp(j2\pi uk/N) \quad k=0,1,\ldots,N-1$$

这时，对应于边界的点数没有改变，但在重构每一个点所需要的计算项大大减少了。如果边界点数很大，M一般选为2的指数次方的整数。

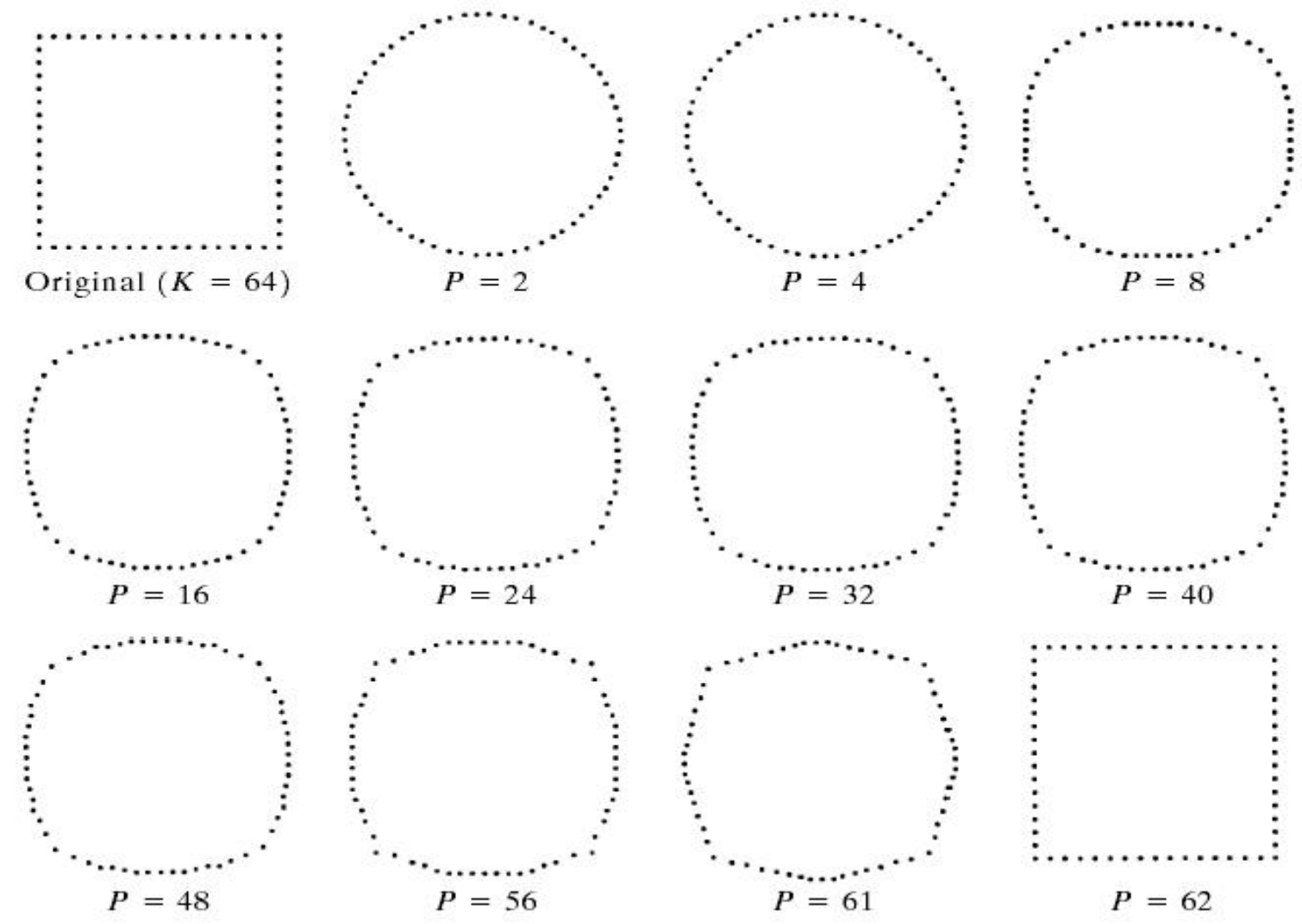
- 傅立叶描述符
  - 2）M的选取与描述符的关系

在上述方法中，相当于对于u > M-1的部分舍去不予计算。由于傅立叶变换中高频部分对应于图像的细节描述，因此M取得越小，细节部分丢失得越多。

只用一些低频分量的傅立叶系数来近似描绘边界形状。



傅立叶描述子



- 傅立叶描述符
  - 3）优点

（1）使用复数作为描述符，对于旋转、平移、放缩等操作和起始点的选取不十分敏感。

（2）以上几何变换均可以通过对描述子函数作简单变换来获得。