数字图像处理

第3章 图像处理中的正交变换

信息科学研究所

3.2 离散余弦变换

图像处理中常用的正交变换除了傅里叶变换外，还有其他一些有用的正交变换。其中离散余弦变换就是一种。离散余弦变换表示为**DCT**。

3.2.1 离散余弦变换的定义

一维离散余弦变换的定义由下式表示

$$F(0)=\frac{1}{\sqrt{N}}\sum_{x=0}^{N-1} f(x) \qquad (3—74)$$

$$F(u)=\sqrt{\frac{2}{N}}\sum_{x=0}^{N-1} f(x)\cos\frac{(2x+1)u\pi}{2N} \qquad (3—75)$$

式中 ***F*(*u*)** 是第 ***u*** 个余弦变换系数，***u*** 是广义频率变量，***u*=1,2, …*N*-1**；***f*(*x*)** 是时域***N***点序列，***x*=0,2, …*N*-1**

一维离散余弦反变换由下式表示

$$f(x)=\sqrt{\frac{1}{N}}F(0)+\sqrt{\frac{2}{N}}\sum_{u=1}^{N-1}F(u)\cos\frac{(2x+1)u\pi}{2N} \qquad \textbf{(3—76)}$$

显然，式(3—74)式(3—75)和式(3—76)构成了一维离散余弦变换对。

二维离散余弦变换的定义由下式表示

$$\begin{aligned}
F(0,0)&=\frac{1}{N}\sum_{x=0}^{N-1}\sum_{y=0}^{N-1} f(x,y)\\
F(0,v)&=\frac{\sqrt{2}}{N}\sum_{x=0}^{N-1}\sum_{y=0}^{N-1} f(x,y)\cdot\cos\frac{(2y+1)v\pi}{2N}\\
F(u,0)&=\frac{\sqrt{2}}{N}\sum_{x=0}^{N-1}\sum_{y=0}^{N-1} f(x,y)\cos\frac{(2x+1)u\pi}{2N}\\
F(u,v)&=\frac{2}{N}\sum_{x=0}^{N-1}\sum_{y=0}^{N-1} f(x,y)\cos\frac{(2x+1)u\pi}{2N}\cdot\cos\frac{(2y+1)v\pi}{2N}
\end{aligned} \qquad \textbf{(3—77)}$$

式(3—77)是正变换公式。其中 $f(x,y)$ 是空间域二维向量之元素。$x, y=0, 1, 2\cdots, N-1$，$F(u,v)$ 是变换系数阵列之元素。式中表示的阵列为 $N\times N$

二维离散余弦反变换由下式表示

$$\begin{aligned} f(x,y) = {} & \frac{1}{N}F(0,0) + \frac{\sqrt{2}}{N}\sum_{v=1}^{N-1} F(0,v)\cos\frac{(2y+1)v\pi}{2N} \\ & + \frac{\sqrt{2}}{N}\sum_{u=1}^{N-1} F(u,0)\cos\frac{(2x+1)u\pi}{2N} \\ & + \frac{2}{N}\sum_{u=1}^{N-1}\sum_{v=1}^{N-1} F(u,v)\cos\frac{(2x+1)u\pi}{2N}\cdot\cos\frac{(2y+1)v\pi}{2N} \end{aligned}$$

(3—78)

式中的符号意义同正变换式一样。式(3—77)和式(3—78)是离散余弦变换的解析式定义。更为简洁的定义方法是采用矩阵式定义。如果令 $N=4$，那么由一维解析式定义可得如下展开式

$$\begin{cases} F(0)=0.500f(0)+0.500f(1)+0.500f(2)+0.500f(3) \\ F(1)=0.653f(0)+0.271f(1)-0.271f(2)-0.653f(3) \\ F(2)=0.500f(0)-0.500f(1)-0.500f(2)+0.500f(3) \\ F(3)=0.271f(0)-0.653f(1)+0.653f(2)-0.271f(3) \end{cases}$$

(3—79)

写成矩阵式

$$\begin{bmatrix} F(0) \\ F(1) \\ F(2) \\ F(3) \end{bmatrix} = \begin{bmatrix} 0.500 & 0.500 & 0.500 & 0.500 \\ 0.653 & 0.271 & -0.271 & -0.653 \\ 0.500 & -0.500 & -0.500 & 0.500 \\ 0.271 & -0.653 & 0.653 & -0.271 \end{bmatrix} \cdot \begin{bmatrix} f(0) \\ f(1) \\ f(2) \\ f(3) \end{bmatrix}$$

(3—80)

若定义 **[A]** 为变换矩阵，**[$F(u)$]** 为变换系数矩阵，**[$f(x)$]** 为时域数据矩阵，则一维离散余弦变换的矩阵定义式可写成如下形式

$$[F(u)] = [A]\,[f(x)]$$ **(3—81)**

同理，可得到反变换展开式

$$\begin{cases} f(0)=0.500F(0)+0.653F(1)+0.500F(2)+0.271F(3) \\ f(1)=0.500F(0)+0.271F(1)-0.500F(2)-0.653F(3) \\ f(2)=0.500F(0)-0.271F(1)-0.500F(2)+0.653F(3) \\ f(3)=0.500F(0)-0.653F(1)+0.500F(2)-0.271F(3) \end{cases}$$

(3—82)

写成矩阵式

$$\begin{bmatrix} f(0) \\ f(1) \\ f(2) \\ f(3) \end{bmatrix} = \begin{bmatrix} 0.500 & 0.653 & 0.500 & 0.271 \\ 0.500 & 0.271 & -0.500 & -0.653 \\ 0.500 & -0.271 & -0.500 & 0.653 \\ 0.500 & -0.653 & 0.500 & -0.271 \end{bmatrix} \cdot \begin{bmatrix} F(0) \\ F(1) \\ F(2) \\ F(3) \end{bmatrix}$$

即 $[f(x)]=[A]'\,[F(u)]$ (3—84)

当然，二维离散余弦变换也可以写成矩阵式

$$\begin{aligned} [F(u,v)] &= [A][f(x,y)][A]' \\ [f(x,y)] &= [A]'[F(u,v)][A] \end{aligned} \quad (3—85)$$

式中**[*f*(*x,y*)]** 是空间数据阵列，***F*[(*u,v*)]** 是变换系数阵列，**[A]** 是变换矩阵，$[A]'$ 是 **[A]** 的转置。

3.2.2 离散余弦变换的正交性

由一维**DCT**的定义可知

$$F(0)=\frac{1}{\sqrt{N}}\sum_{x=0}^{N-1} f(x)$$

$$F(u)=\sqrt{\frac{2}{N}}\sum_{x=0}^{N-1} f(x)\cos\frac{(2x+1)u\pi}{2N}$$

它的基向量是

$$\left\{\sqrt{\frac{1}{N}},\ \sqrt{\frac{2}{N}}\cos\frac{(2x+1)u\pi}{2N}\right\} \quad (3—86)$$

在高等数学中，切比雪夫多项式的定义为

$$T_0(p)=\sqrt{\frac{1}{N}}$$

$$T_u(z_x)=\sqrt{\frac{2}{N}}\cos[u\arccos(z_x)] \quad \textbf{(3—87)}$$

式中 $T_u(z_x)$ 是 u 和 Z_x 多项式。它的第**N**个多项式为

$$T_N(z_x)=\sqrt{\frac{2}{N}}\cos[N\arccos(z_x)]$$

如果 $T_N(z_x)=0$

那么 $z_x=\cos\frac{(2x+1)\pi}{2N}$

将此式代入 $T_N(z_x)$

$$T_N=\sqrt{\frac{2}{N}}\cos\left\{u\arccos\left[\cos\frac{(2x+1)\pi}{2N}\right]\right\}$$
$$=\sqrt{\frac{2}{N}}\cos\frac{(2x+1)u\pi}{2N} \qquad (3—88)$$

比较一下余弦变换的基向量

$$\left\{\sqrt{\frac{1}{N}},\ \sqrt{\frac{2}{N}}\cos\frac{(2x+1)u\pi}{2N}\right\}$$

显然，这与一维**DCT**的基向量是一致的。因为切比雪夫多项式是正交的，所以**DCT**也是正交的。另外，离散余弦变换的正交性也可以通过实例看出。如前所示，当**N**= 4 时，

$$[A]=\begin{bmatrix}0.500 & 0.500 & 0.500 & 0.500\\ 0.653 & 0.271 & -0.271 & -0.653\\ 0.500 & -0.500 & -0.500 & 0.500\\ 0.271 & -0.653 & 0.653 & -0.271\end{bmatrix}$$

$$[A]'=\begin{bmatrix}0.500 & 0.635 & 0.500 & 0.2710\\ 0.500 & 0.271 & -0.500 & -0.653\\ 0.500 & -0.271 & -0.500 & 0.653\\ 0.500 & -0.653 & 0.500 & -0.271\end{bmatrix}$$

显然 $[A][A]'=[I]$

这是满足正交条件的。从上述讨论可见，离散余弦变换是一类正交变换。

3.2.3 离散余弦变换的计算

与傅里叶变换一样，离散余弦变换自然可以由定义式出发进行计算。但这样的计算量太大，在实际应用中很不方便。所以也要寻求一种快速算法。

首先，从定义出发，作如下推导

$$F(u)=\sqrt{\frac{2}{N}}\sum_{x=0}^{N-1}f(x)\cos\frac{(2x+1)u\pi}{2N}$$
$$=\sqrt{\frac{2}{N}}\sum_{x=0}^{N-1}f(x)R_e\left\{e^{-j\frac{(2x+1)u\pi}{2N}}\right\}$$
$$=\sqrt{\frac{2}{N}}R_e\left\{\sum_{x=0}^{N-1}f(x)e^{-j\frac{(2x+1)u\pi}{2N}}\right\}$$

(3—89)

式中 R_e 是取其实部的意思。如果把时域数据向量作下列延拓，即：

$$f_e(x)=\begin{cases} f(x) & x=0,1,2,\cdots\cdots,N-1 \\ 0 & x=N,N+1,\cdots\cdots,2N-1 \end{cases} \qquad (3—90)$$

则 $f_e(x)$ 的离散余弦变换可写成下式

$$F(0)=\frac{1}{\sqrt{N}}\sum_{x=0}^{2N-1} f_e(x)$$

$$F(u)=\sqrt{\frac{2}{N}}\sum_{x=0}^{2N-1} f_e(x)\cos\frac{(2x+1)u\pi}{2N}$$

$$=\sqrt{\frac{2}{N}}R_e\left\{\sum_{x=0}^{2N-1} f_e(x)\,e^{-j\frac{(2x+1)u\pi}{2N}}\right\} \qquad (3—91)$$

$$=\sqrt{\frac{2}{N}}R_e\left\{e^{-j\frac{u\pi}{2N}}\cdot\sum_{x=0}^{2N-1} f_e(x)\,e^{-j\frac{2xu\pi}{2N}}\right\}$$

由式**(3—91)**可见

$$\sum_{x=0}^{2N-1} f_e(x)\ e^{-j\frac{2xu\pi}{2N}}$$

是**2*N***点的离散傅里叶变换。

所以，在作离散余弦变换时，可以把序列长度延拓为**2*N***，然后作离散傅里叶变换，产生的结果取其实部便可得到余弦变换。

同样道理，在作反变换时，首先在变换空间，把 **[*F*(*u*)]** 作如下下延拓

$$F_e(u)=\begin{cases} F(u) & u=0,1,2,\cdots\cdots,N-1 \\ 0 & u=N,N+1,\cdots\cdots,2N-1 \end{cases} \qquad (3—92)$$

那么，反变换也可用式**(3—93)**表示

$$f(x)=\frac{1}{\sqrt{N}}F_e(0)+\sqrt{\frac{2}{N}}\sum_{u=1}^{2N-1}F_e(u)\cos\frac{(2x+1)u\pi}{2N}$$

$$=\frac{1}{\sqrt{N}}F_e(0)+\sqrt{\frac{2}{N}}\sum_{u=1}^{2N-1}F_e(u)R_e\left\{e^{j\frac{(2x+1)u\pi}{2N}}\right\}$$

$$=\frac{1}{\sqrt{N}}F_e(0)+\sqrt{\frac{2}{N}}\sum_{u=1}^{2N-1}F_e(u)R_e\left\{e^{j\frac{2xu\pi}{2N}}\cdot e^{j\frac{u\pi}{2N}}\right\}$$

$$=\frac{1}{\sqrt{N}}F_e(0)+\sqrt{\frac{2}{N}}R_e\left\{\sum_{u=1}^{2N-1}F_e(u)\cdot e^{j\frac{u\pi}{2N}}\cdot e^{j\frac{2xu\pi}{2N}}\right\}$$

$$=\left[\frac{1}{\sqrt{N}}-\sqrt{\frac{2}{N}}\right]F_e(0)+\sqrt{\frac{2}{N}}R_e\left\{\sum_{u=0}^{2N-1}\left[F_e(u)\cdot e^{j\frac{u\pi}{2N}}\right]e^{j\frac{2xu\pi}{2N}}\right\}$$

(3—93)

由式(3—93)可见，离散余弦反变换可以从 $\left[F_e(u) \cdot e^{j\frac{u\pi}{2N}} \right]$ 的2N点反傅里叶变换实现。

离散余弦变换的快速算法：

利用 ***FFT*** 解决离散余弦变换的快速运算问题。

离散傅里叶变换和余弦变换在快速算法中都要用到复数乘法，占用的时间仍然比较多。在某些应用领域中，需要更为便利更为有效的变换方法。沃尔什变换就是其中的一种。

沃尔什函数是在**1923**年由美国数学家沃尔什**(J.L.Walsh)**提出来的。在沃尔什的原始论文中，给出了沃尔什函数的递推公式，这个公式是按照函数的序数(正交区间内过零点平均数)来定义的。

不久以后，这种规定函数序数的方法也被波兰数学家卡兹马兹**(S.Kaczmarz)**采用了，所以，通常将这种规定函数序数的方法称为沃尔什一卡兹马兹**(Walshi-Kaczmarz)**定序法。

1931年美国数学家佩利**(R.E.A.C.Paley)**又给沃尔什函数提出了一个新的定义。他指出，沃尔什函数可以用有限个拉德梅克**(Rademacher)**函数的乘积来表示。

这样得到的函数的序数与沃尔什得到的函数的序数完全不同。这种定序方法是用二进制来定序的，所以称为

二进制序数或自然序数。

利用只包含＋1和－1的正交矩阵可以将沃尔什函数表示为矩阵形式。早在1867年，英国数学家希尔威斯特(J.J.Sylvester)已经研究过这种矩阵。后来，法国数学家哈达玛(M.Hadamard)在1893年将这种矩阵加以普遍化，建立了所谓哈达玛矩阵。

利用克罗内克乘积算子(Kronecker Product Operator)不难把沃尔什函数表示为哈达玛矩阵形式。利用这种形式定义的沃尔什函数称为克罗内克序数。这就是沃尔什函数的第三种定序法。

由上述历史可见，沃尔什函数及其有关函数的数学基础早已奠定了。但是，这些函数在工程中得到应用却是近几十年的事情。主要原因是由于半导体器件和计算机在近几十年得到迅速发展，它们的发展为沃尔什函数的实用解决了手段问题，因此，也使沃尔什函数得到了进一步发展。

与傅里叶变换相比，沃尔什变换的主要优点在于存储空间少和运算速度高，这一点对图像处理来说是至关重要的，特别是在大量数据需要进行实时处理时，沃尔什函数就更加显示出它的优越性。

3.3.1 拉德梅克函数

拉德梅克(Rademacher)函数集是一个不完备的正交函数集，由它可以构成完备的沃尔什函数。在这里首先介绍一下拉德梅克函数。拉德梅克函数包括n和t两个自变量，用$R(n,t)$来表示拉德梅克函数。它可用下式来表示

$$R(n,t) = \mathrm{sgn}(\sin\ 2^{n}\pi t) \qquad (3—100)$$

$$sgn\,(x) = \begin{cases} 1 & x > 0 \\ -1 & x < 0 \end{cases} \qquad (3—101)$$

当**x=0**时， ***sgn*(*x*)** 无定义。

由***sin***函数的周期性知道***R*(*n*,*t*)**也是周期性函数。由式**(3—100)**可见，

当 ***n*=1**时，***R*(1,*t*)**的周期为**1**；

当 ***n*=2**时，***R*(2,*t*)**的周期为**1/2**；

当 ***n*=3**时，***R*(3,*t*)**的周期为 $\frac{1}{2^{2}}$ ；

一般情况下可用下式表示

$$R(n,t) = R\left(n, t + \frac{1}{2^{n-1}}\right) \qquad n = 1,2,\cdots\cdots$$

(3—102)

拉德梅克函数的波形如图**3—9**所示。

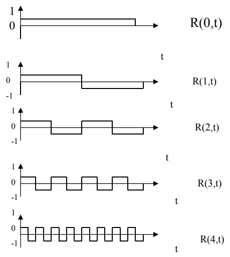

图 **3—9** 拉德梅克函数

由图**3—9**可见，拉德梅克函数有如下一些规律：

（１）***R*(*n*,*t*)**的取值只有**+1**和**-1**。

（２）***R*(*n*,*t*)**是***R*(*n-1*,*t*)**的二倍频。因此，如果已知最高次数***m*=*n***，则其他拉德梅克函数可由脉冲分频器来产生。

（**3**）如果已知***n***，那么**,** ***R*(*n*,*t*)**有 2^{n-1} 个周期，其中**0<*t*<1**；

（**4**）如果在 $t = \frac{k + \frac{1}{2}}{2^{N}}$ 处作取样， 则可得到一数据序列***R*(*n*,*k*)**，$\mathrm{k} = 0,1,2,\cdots\cdots,2^{n} - 1$ 。每一取样序列将与下述矩阵相对应。

这里我们取***n*=3**，***k*=0,1,2,......7**。

$$\begin{bmatrix} R(0,k) \\ R(1,k) \\ R(2,k) \\ R(3,k) \end{bmatrix} \Leftrightarrow \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 \\ 1 & 1 & -1 & -1 & 1 & 1 & -1 & -1 \\ 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 \end{bmatrix} \quad (3—103)$$

采用上述离散矩阵形式就可以用计算机进行灵活处理。

3.3.2 沃尔什函数

沃尔什函数系是完备的正交函数系，其值也是只取＋1和－1。从排列次序来定义不外乎三种：

第一种是按沃尔什排列或称按列率排列来定义；

第二种是按佩利排列定义 （自然序数）；

第三种是按哈达玛排列来定义（第三定序法）。

还可用其它方式来定义，但沃尔什函数的定义至今尚未统一，下面分别讨论上述三种排列方法定义的沃尔什函数。

3.3.1 按沃尔什排列的沃尔什函数

按沃尔什排列的沃尔什函数用 $wal_W(i,t)$ 来表示。

函数波形如图3—10所示。

按沃尔什排列的沃尔什函数实际上就是按列率排列的沃尔什函数。

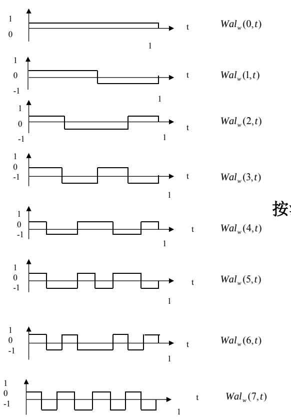

图 3—10
按沃尔什排列的沃尔什函数

从波形上可总结出如下规律：

（1）在 $wal_W(i,t)$ 中，i 就是波形在正交区间的变号次数；

如：$wal_w(0,t)$ 变号次数为0；

$wal_w(1,t)$ 变号次数为1；

$wal_w(2,t)$ 变号次数为2；

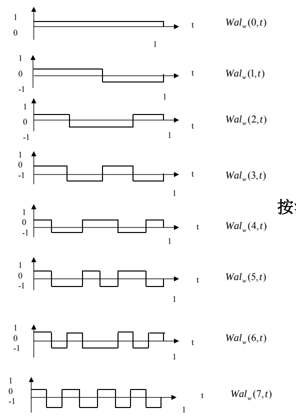

图 3—10
按沃尔什排列的沃尔什函数

（2）列率：通常把正交区间内波形变号次数的二分之一称为列率(*sequency*)。如果令 *i*为波形在正交区间内的变号次数，那么，按照*i*为奇数或偶数，函数 $wal_W(i,t)$ 的列率将分别由下式来决定

$$S_i = \begin{cases} 0 & i = 0 \\ \dfrac{i+1}{2} & i = odd \\ \dfrac{i}{2} & i = even \end{cases}$$

（3）按沃尔什排列的沃尔什函数可由拉德梅克函数构成，它的表达式如下

$$wal_W(i,t) = \prod_{k=0}^{p-1} \left[R(k+1,t)\right]^{g(i)_k}$$
$$g(i)_k \in \{0,1\} \qquad (3—105)$$

式中**$R(k+1,t)$**是拉德梅克函数，**$g(i)$**是为***i***的格雷码，**$g(i)_k$**是此格雷码的第***k***位数字，***p***为格雷码长度。

一个正整数可以编成自然二进码，但也可以编成格雷码。格雷码也称为反射码。格雷码的特点是：两个相邻数的格雷码只有一个码位的值不同。例如：

2的格雷码是 **(0 0 1 1)**，

3的格雷码为 **(0 0 1 0)**。

这两个相邻的数字的格雷码只有第四个码位的值不同。

在脉冲编码技术中，常常采用这种码，以便得到较好的误差特性。一个正整数的自然二进码和格雷码之间是可以互相转换的。从自然二进码转成格雷码的方法如下：

设一个十进制数的自然二进码为：

$n = \left(n_{p-1} n_{p-2} \cdots\cdots n_k \cdots\cdots n_2 n_1 n_0\right)_B$

并设该数的格雷码为：

$g = \left(g_{p-1} g_{p-2} \cdots\cdots g_k \cdots\cdots g_2 g_1 g_0\right)_G$

其中 n_k 和 g_k 分别为自然二进码和格雷码内的码位数字，并且 n_k 、$g_k \in \{0,1\}$ 。它们之间的关系可用式**(3—106)**表示

$$\begin{cases} g_{p-1} = n_{p-1} \\ g_{p-2} = n_{p-1} \oplus n_{p-2} \\ g_{p-3} = n_{p-2} \oplus n_{p-3} \\ \cdots \\ g_k = n_{k-1} \oplus n_k \\ \cdots \\ g_1 = n_2 \oplus n_1 \\ g_0 = n_1 \oplus n_0 \end{cases} \qquad (3—106)$$

式中 $\oplus$ 代表模 2 加。

模2加的规律：

$1 \oplus 1 = 0$

$1 \oplus 0 = 1$

$0 \oplus 1 = 1$

$0 \oplus 0 = 0$

例： **P=4** 试求（**2**）的格雷码。

解： $n(2) = (0010)_B$

其中：

$n_3 = 0$

$n_2 = 0$

$n_1 = 1$

$n_0 = 0$

$g_3 = n_3 = 0$

$g_2 = n_3 \oplus n_2 = 0 \oplus 0 = 0$

$g_1 = n_2 \oplus n_1 = 0 \oplus 1 = 1$

$g_0 = n_1 \oplus n_0 = 1 \oplus 0 = 1$

其格雷码 $g(2) = (0011)_g$

同理 若 $n(3) = (0011)_B$ 则其格雷码为

$g(3) = (0010)_g$

在格雷码中，有如下关系存在：

$$g(m) \oplus g(n) = g(m \oplus n) \qquad (3—107)$$

例：设： $m = 2 = (0010)_B$
$n = 3 = (0011)_B$
$g(2) = (0011)_g$
$g(3) = (0010)_g$

则 $g(2)\oplus g(3) = (0001)$

而 $(2)_B \oplus (3)_B = (0001)$

所以： $g(2)\oplus g(3) = g(2\oplus 3)$

从正整数的格雷码也可以求出该十进数的自然二进码。其转换方法如下：

设正整数的格雷码为：

$$g(n) = (g_{p-1}g_{p-2}g_{p-3}\cdots\cdots g_k\cdots\cdots g_2g_1g_0)$$

又设其自然二进码为：

$$B(n) = (n_{p-1}n_{p-2}n_{p-3}\cdots\cdots n_k\cdots\cdots n_2n_1n_0)$$

则

$$\begin{cases} n_{p-1} = g_{p-1} \\ n_{p-2} = g_{p-1}\oplus g_{p-2} \\ n_{p-3} = g_{p-1}\oplus g_{p-2}\oplus g_{p-3} \\ \cdots \\ n_k = g_{p-1}\oplus g_{p-2}\oplus g_{p-3}\oplus\cdots\cdots\oplus g_k \\ \cdots \\ n_2 = g_{p-1}\oplus g_{p-2}\oplus g_{p-3}\oplus\cdots\cdots\oplus g_2 \\ n_1 = g_{p-1}\oplus g_{p-2}\oplus g_{p-3}\oplus\cdots\cdots\oplus g_2\oplus g_1 \\ n_0 = g_{p-1}\oplus g_{p-2}\oplus g_{p-3}\oplus\cdots\cdots\oplus g_2\oplus g_1\oplus g_0 \end{cases} \tag{3—108}$$

例：***n*的格雷码为1011**，求其自然二进码表示。

由给定的格雷码可知：$(n)_g = (1011)$

其中：

$g_3 = 1$
$g_2 = 0$
$g_1 = 1$
$g_0 = 1$

所以：

$n_3 = g_3 = 1$
$n_2 = g_3\oplus g_2 = 1\oplus 0 = 1$
$n_1 = g_3\oplus g_2\oplus g_1 = 1\oplus 0\oplus 1 = 0$
$n_0 = g_3\oplus g_2\oplus g_1\oplus g_0 = 1\oplus 0\oplus 1\oplus 1 = 1$

即：自然二进码为 $(1101)_B$

以上便是格雷码的定义及格雷码与自然二进码之间的转换方法。

例： 用公式**(3—105)**求 ***p*=4** 时的 $wal_w(5,t)$ 。

解： 因为***i*=5**，所以**5**的自然二进码为**(0101)**。由前面所述的转换规则可得到格雷码为**(0111)**。因此，有下面的对应关系

$g(5)_3 \quad g(5)_2 \quad g(5)_1 \quad g(5)_0$

$(0 \quad 1 \quad 1 \quad 1)$

↑ ↑ ↑ ↑

第3位 第2位 第1位 第0位

即

$$g(5)_0 = 1,\ g(5)_1 = 1$$
$$g(5)_2 = 1,\ g(5)_3 = 0$$

代入式**(3—105)**得

$$wal_w(i,t) = \prod_{k=0}^{p-1} \left[R(k+1,t)\right]^{g(i)_k}$$

$$wal_w(5,t) = \left[R(1,t)\right]^1 \cdot \left[R(2,t)\right]^1 \cdot \left[R(3,t)\right]^1 \cdot \left[R(4,t)\right]^0$$

$$= R(1,t) \cdot R(2,t) \cdot R(3,t)$$

3.3.2 按佩利排列的沃尔什函数

用 $wal_p(i,t)$ 来表示按佩利排列的沃尔什函数，其波形如图**3—11**所示。

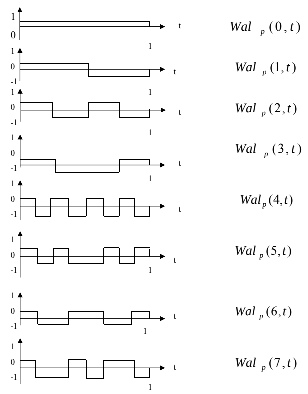

图 **3—11** 按佩利排列的沃尔什函数

按佩利排列的沃尔什函数也可以由拉德梅克函数产生。其定义由下式表示

$$wal_P(i,t) = \prod_{k=0}^{p-1} \left[R(k+1,t)\right]^{i_k} \qquad \textbf{(3—111)}$$

式中 $R(k+1,t)$ 是拉德梅克函数， $i_k \in \{0,1\}$ 是将函数序号写成自然二进码的第***k***位数字 i_k。即：

$$(i) = (i_{n-1} i_{n-2} \cdots \cdots i_2 i_1 i_0)_B$$

例：***p***=**3**时，求 $wal_p(1,t)$

因为***i***=**1**，所以自然二进码为：

$[0 \quad 0 \quad 1]$

↑ ↑ ↑

第2位 第1位 第0位

代入式**(3—111)**得：

$$wal_P(1,t) = \prod_{k=0}^{2} \left[R(k+1,t)\right]^{i_k}$$
$$= [R(1,t)]^1 \cdot [R(2,t)]^0 \cdot [R(3,t)]^0 \cdot = R(1,t)$$

例：p=3，求 $wal_p(5,t)$ 。

因为i=5，所以，自然二进码为(101)，代入式(3—111)，则

$$wal_p(5,t)=\prod_{k=0}^{2}[R(k+1,t)]^{i_k}$$
$$=[R(1,t)]^1\cdot[R(2,t)]^0\cdot[R(3,t)0]^1\cdot=R(1,t)\cdot R(3,t)$$

当p =3时的8个沃尔什函数经取样后可得矩阵$H_p(3)$ 如式(3—112)

$$H_p(3)=\begin{bmatrix}1&1&1&1&1&1&1&1\\1&1&1&1&-1&-1&-1&-1\\1&1&-1&-1&1&1&-1&-1\\1&1&-1&-1&-1&-1&1&1\\1&-1&1&-1&1&-1&1&-1\\1&-1&1&-1&-1&1&-1&1\\1&-1&-1&1&1&-1&-1&1\\1&-1&-1&1&-1&1&1&-1\end{bmatrix}$$

(3—112)

由按佩利排列的沃尔什函数前8个波形的规律：

(1)、函数序号i与正交区间内取值符号变化次数有表3—1所列之关系。

表3.1 按佩利排列的沃尔什函数序号与变号次数的关系

i	0	1	2	3	4	5	6	7
变号数	0	1	3	2	7	6	4	5

(2)、i与变号次数的关系是自然二进码与格雷码的关系。如i=6=$(110)_B$，这个自然二进制码按格雷码读出是4，也就是说，把十进制数变成自然二进码，然后按格雷码的规律反变回十进制数，这个数就是这个序号的沃尔什函数的变号次数。

3.3.3 按哈达玛排列的沃尔什函数

按哈达玛排列的沃尔什函数是从 2^n 阶哈达玛矩阵得来的。2^n 阶哈达玛矩阵每一行的符号变化规律，对应某个沃尔什函数在正交区间内符号变化的规律，也就是说，2^n 阶哈达玛矩阵的每一行就对应着一个离散沃尔什函数。2^n 阶哈达玛矩阵有如下形式

$$H(0)=[1] \qquad (3—113)$$

$$H(1)=\begin{bmatrix}1&1\\1&-1\end{bmatrix} \qquad (3—114)$$

$$H(2)=\begin{bmatrix}1&1&1&1\\1&1&1&-1\\1&1&-1&-1\\1&-1&-1&1\end{bmatrix} \qquad (3—115)$$

一般情况

$$H(m)=\begin{bmatrix} H(m-1) & H(m-1) \\ H(m-1) & -H(m-1) \end{bmatrix}=H(m-1)\otimes H(1)$$

(3—116)

式(3—116)是哈达玛矩阵的递推关系式。利用这个关系式可以产生任意 2^n 阶哈达玛矩阵。这个关系也叫做克罗内克积(Kronecker Product)关系，或叫直积关系。

按哈达玛排列的沃尔什函数用 $wal_h(i,t)$ 来表示。它的前八个函数波形如图3—12所示。按哈达玛排列的沃尔什函数也可以写成矩阵式

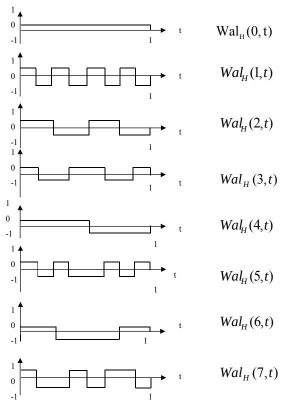

图 3—12 按哈达玛排列的沃尔什函数

$$H_h(3)=\begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 & 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 \\ 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 \\ 1 & -1 & 1 & -1 & -1 & 1 & -1 & 1 \\ 1 & 1 & -1 & -1 & -1 & -1 & 1 & 1 \\ 1 & -1 & -1 & 1 & -1 & 1 & 1 & -1 \end{bmatrix}$$

(3—117)

按哈达玛排列的沃尔什函数有如下一些特点：

(1)、从**2**阶哈达玛矩阵可得到**2**个沃尔什函数，从**4**阶哈达玛矩阵可得到**4**个沃尔什函数，一般地说，2^n 阶哈达玛矩阵可得到 2^n 个沃尔什函数。

(2)、由不同阶数的哈达玛矩阵得到的沃尔什函数排列顺序是不同的。例如，从 $H_h(4)$ 得到的沃尔什函数 $wal_h(2,t)$ 并不是从 $H_h(8)$ 得到的 $wal_h(2,t)$，而是从 $H_h(8)$ 得到的 $wal_h(4,t)$。

(3)、由于哈达玛矩阵的简单的递推关系，使得按哈达玛排列的沃尔什函数特别容易记忆。

(4)、按哈达玛排列的沃尔什函数也可以由拉德梅克函数产生，解析式如式**(3—118)**所示。

$$wal_h(i,t)=\prod_{k=0}^{p-1}\left[R(k+1,t)\right]^{\langle i_k\rangle} \qquad \textbf{(3—118)}$$

式中 $R(k+1,t)$ 仍然是拉德梅克函数，< i_k >是把***i***的自然二进码反写后的第***k***位数字，并且 $i_k\in\{0,1\}$ ，也就是

$$(i)=(i_{n-1}i_{n-2}\cdots\cdots i_2 i_1 i_0)$$

反写后 $\langle i\rangle=(i_0\ i_1\ i_2\cdots\cdots i_{n-2}\ \ i_{n-1})$

…… ↑ ↑

第1位第0位

例如：求***p*=3**时，$wal_h(6,t)$ 的波形。$2^n=2^3=8$

第一种方法可用比较简单的方法是写出

阶哈达玛矩阵 $H_h(8)$ ，并自上而下从**0**数起至第**6**行就是 $wal_h(6,t)$ 。

第二种方法是应用数学解析式。因为

$i=6=(110)_B$ ，所以：

$$\langle i\rangle=(\underset{\langle i_2\rangle}{\underset{\uparrow}{0}}\quad \underset{\langle i_1\rangle}{\underset{\uparrow}{1}}\quad \underset{\langle i_0\rangle}{\underset{\uparrow}{1}})$$

代入式**(3—118)**得

$$wal_h(6,t)=[R(1,t)]^{\langle i_0\rangle}\cdot[R(2,t)]^{\langle i_1\rangle}\cdot[R(3,t)]^{\langle i_2\rangle}=R(1,t)\cdot R(2,t)$$

三种定义下的沃尔什函数，尽管它们的排列顺序各不相同，但三种排序方法得到的沃尔什函数是有一定关系的。它们之间的关系如表**3—2**和图**3—13**所示。

表3 -2 三种排列的前 8 个沃尔什函数之间的关系表

$wal_h(i,t)$	$wal_w(i,t)$	$wal_p(i,t)$
$wal_h(0,t)$	$wal_w(0,t)$	$wal_p(0,t)$
$wal_h(1,t)$	$wal_w(7,t)$	$wal_p(4,t)$
$wal_h(2,t)$	$wal_w(3,t)$	$wal_p(2,t)$
$wal_h(3,t)$	$wal_w(4,t)$	$wal_p(6,t)$
$wal_h(4,t)$	$wal_w(1,t)$	$wal_p(1,t)$
$wal_h(5,t)$	$wal_w(6,t)$	$wal_p(5,t)$
$wal_h(6,t)$	$wal_w(2,t)$	$wal_p(3,t)$
$wal_h(7,t)$	$wal_w(5,t)$	$wal_p(7,t)$

例如 $Wal_p(2,t)$ 的 $wal_w(i,t)$ 和 $wal_H(i,t)$ 间的关系如下：$2=(010)_B$，按格雷码读，即$(010)_G=3$，所以就是 $wal_w(3,t)$，(010)比特倒置后为(010)，按二进码读仍为2，则 $wal_p(2,t)$ 就是 $wal_H(2,t)$ 。其他以次类推。以上就是沃尔什函数三种定义之间的关系。

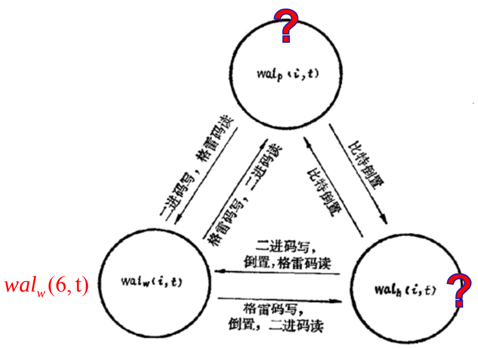

图 3 —13 三种定义的沃尔什函数序号间的关系