

⑥ 30-34

# 基于具有对称生成元正交小波的图象重构算法

TN911.1

TN919.8

刘一武 张邦礼<sup>✓</sup> 王光学

(重庆大学自动化系, 重庆, 400044; 第一作者 29岁, 男, 博士生)

**摘要** 针对具有对称生成元的正交小波基, 利用其正交镜像滤波器的性质, 提出了一种新的图象重构算法。理论分析和计算机仿真结果表明, 本算法不但显著地提高了图象重构运算的速度, 并且使图象重构的质量也得到一定的改善。

**关键词** 图象变换; 滤波器; 算法 / 正交小波

中国图书资料分类法分类号 TN919.8

## 0 引言

对称生成元

近年来, 小波变换因其良好的时频局部性特点, 已经成为图象压缩编码的研究热点<sup>[1~3]</sup>。小波变换图象编码方法有效地克服了传统 DCT 方法中的方块效应和振铃效应, 且更适合人类视觉系统(HVS)的特点和图象数据压缩的需要<sup>[4]</sup>, 是一种很有前途的编码方法。

但是, 在可视电话、会议电视、视频通信等诸多应用方面, 小波变换方法尚远不能满足实时性要求。目前通用的图象小波变换及重构的快速算法是 Mallat 算法<sup>[1]</sup>, 如何进一步提高小波变换及重构的速度, 将成为小波方法能否广泛应用的关键。

笔者针对一大类生成元(尺度函数)具有对称性的正交小波基, 利用其滤波器对称性质, 在 Mallat 算法基础上提出一种新的重构方法, 该方法能够大幅度减少图象重构的运算量。

## 1 图象小波变换系数的重构公式及 Mallat 算法实现

对图象进行二维小波变换, 得到四幅子带图象,  $C_0$  表示图象纵横两向的低频信息,  $D_1, D_2, D_3$  分别对应不同方向的高频信息。在更粗糙一级的分辨率上,  $C_0$  又可分解为  $C_i, D_i (i=1, 2, 3), \dots$ 。图1是图象经过两级分辨率分解后的子带划分情况。

图象重构是上述多分辨率分解的逆过程。设  $C_i, D_i (i=1, 2, 3)$  是对应尺度  $2^{-i}$  的四个大小为  $N \times M$  的子块, 则它们可以重构尺度

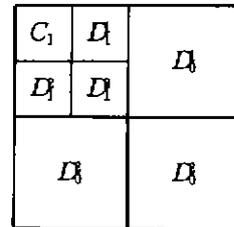


图1 图象小波分解的频域划分

• 收文日期 1997-11-21  
中国工程物理研究院院外基金资助项目

$2^{-(k-1)}$  下的低频分量  $C_{k-1}$ , 公式为

$$C_{k-1,n,m} = 2 \left\{ \sum_{j,l} c_{k,j,l} h_{n-2j} h_{m-2l} + \sum_{j,l} d_{k,j,l} h_{n-2j} g_{m-2l} \right. \\ \left. + \sum_{j,l} \bar{d}_{k,j,l} g_{n-2j} h_{m-2l} + \sum_{j,l} \bar{c}_{k,j,l} g_{n-2j} g_{m-2l} \right\} \\ n = 0, 1, \dots, 2N-1; m = 0, 1, \dots, 2M-1 \quad (1)$$

式中  $C_{k,n,m}$  表示子块  $C_k$  的第  $(n, m)$  系数,  $d_{k,n,m}$  表示子块  $D_k$  的第  $(n, m)$  系数,  $i = 1, 2, 3$ .  $h_n, g_n$  分别表示所用小波对应的离散低通滤波器和带通滤波器系数。

Mallat 算法实现式(1)时, 先将  $C_k, D_k (i = 1, 2, 3)$  各行间隔插零, 扩展成  $2N$  长度, 分别进行列滤波; 再将各行间隔插零, 扩展成  $2M$  长度, 分别进行行滤波。不妨设每次滤波计算为  $K$  个单位, 则 Mallat 算法实现式(1)的计算量大约为  $16MNK$ 。

## 2 基于对称正交镜像滤波器的快速重构算法

正交镜像滤波器 (Quadrature mirror filters) 具有如下性质

$$g_n = (-1)^{1-n} h_{1-n}$$

生成元具有对称性的正交小波基, 其低通滤波器亦具有对称性, 即

$$h_{-n} = h_n$$

为了下文表述方便, 令  $\bar{h}_n = h_{-n}, \bar{g}_n = g_{-n}$ .  $H, G$  分别表示由  $h_n, \bar{g}_n$  构成的向量

$$H = [h_{-L_0}, \dots, h_0, \dots, h_{L_0}]^T$$

$$G = [\bar{g}_{-L_0-1}, \dots, \bar{g}_0, \dots, \bar{g}_{L_0-1}]^T$$

滤波器长度  $L = 2L_0 + 1$ .

### 2.1 重构公式的矩阵表示

$$\text{令 } x_{2j, 2l} = c_{k,j,l}, \quad x_{2j+1, 2l} = x_{2j, 2l+1} = x_{2j+1, 2l+1} = 0$$

矩阵

$$X(n, m) = \begin{bmatrix} \vdots & & \\ \cdots & x_{n,m} & \cdots \\ \vdots & & \end{bmatrix}_{L \times L}$$

表示中心为  $(n, m)$  的  $L \times L$  矩阵。

重构公式(1)中

$$\sum_{j,l} c_{k,j,l} h_{n-2j} h_{m-2l} = \sum_{j=0}^{N-1} \sum_{l=0}^{M-1} c_{k,j,l} h_{n-2j} h_{m-2l} = \sum_{j=0}^{2N-1} \sum_{l=0}^{2M-1} x_{j,l} h_{n-j} h_{m-l} \\ = \sum_{j=-n}^{2N-1-n} \sum_{l=-m}^{2M-1-m} x_{j+n,l+m} \bar{h}_j \bar{h}_l = \sum_{j=-L_0}^{L_0} \sum_{l=-L_0}^{L_0} x_{j+n,l+m} \bar{h}_j \bar{h}_l \quad (\text{进行边界延拓}) \\ = H^T X(n, m) H \quad (2a)$$

同理, 记

$$Y^i(n, m) = \begin{bmatrix} \vdots \\ \cdots & y_{k,m}^i & \cdots \\ \vdots \end{bmatrix}_{L \times L} \quad i = 1, 2, 3$$

表示中心为  $(n, m)$  的  $L \times L$  矩阵, 其中

$$y_{2j, 2l}^i = d_{k, j, l}^i \quad y_{2j+1, 2l}^i = y_{2j, 2l+1}^i = y_{2j+1, 2l+1}^i = 0 \quad i = 1, 2, 3$$

可以得到

$$\sum_{j, l} d_{k, j, l}^i h_{n-2j} g_{m-2l} = H^T Y^i(n, m-1) G \quad (2b)$$

$$\sum_{j, l} d_{k, j, l}^i g_{n-2j} h_{m-2l} = G^T Y^i(n-1, m) H \quad (2c)$$

$$\sum_{j, l} d_{k, j, l}^i g_{n-2j} g_{m-2l} = G^T Y^i(n-1, m-1) G \quad (2d)$$

## 2.2 重构算法推导

记  $H_1 = (H + G)/2 \quad H_2 = (H - G)/2$

则

$$H = H_1 + H_2 \quad G = H_1 - H_2 \quad (3)$$

根据滤波器的性质,  $H_1, H_2$  都是间隔为 0 的向量。

将(2)、(3)代入(1), 有

$$\begin{aligned} c_{k-1, n, m}/2 = & H_1^T [X(n, m) + Y^1(n, m-1) + Y^2(n-1, m) + Y^3(n-1, m-1)] H_1 \\ & + H_1^T [X(n, m) - Y^1(n, m-1) + Y^2(n-1, m) - Y^3(n-1, m-1)] H_2 \\ & + H_2^T [X(n, m) + Y^1(n, m-1) - Y^2(n-1, m) - Y^3(n-1, m-1)] H_1 \\ & + H_2^T [X(n, m) - Y^1(n, m-1) - Y^2(n-1, m) + Y^3(n-1, m-1)] H_2 \end{aligned} \quad (4)$$

在四项矩阵求和差中, 由于  $X, Y^i$  分别由  $C_k, D_k$  间隔插零所得, 实际上各矩阵和形如

$$\begin{bmatrix} c & d & c & d & \cdots \\ d & d & d & d & \\ c & d & c & d & \\ d & d & d & d & \\ \vdots & & & & \end{bmatrix}$$

### 1) 列滤波

在列滤波时, 只需计算

$$\begin{aligned} a_{n, m} = & H_1^T [X(n, m) + Y^1(n, m-1) + Y^2(n-1, m) + Y^3(n-1, m-1)] e \\ b_{n, m} = & H_2^T [X(n, m) + Y^1(n, m-1) + Y^2(n-1, m) + Y^3(n-1, m-1)] e \\ & n = 0, 1, \dots, 2N-1 \end{aligned} \quad (5)$$

其中  $e = [1, 0, \dots, 0]^T$ .

将  $H_1, H_2$  剔除零值得到向量  $V_1, V_2$ , 则

$$\begin{aligned} a_{2n, 2m} &= V_1^T c_k(n, m) & a_{2n, 2m+1} &= V_1^T d_k(n, m) \\ a_{2n+1, 2m} &= V_1^T d_k(n, m) & a_{2n+1, 2m+1} &= V_1^T d_k(n, m) \\ b_{2n, 2m} &= V_2^T d_k(n, m) & b_{2n, 2m+1} &= V_2^T d_k(n, m) \\ b_{2n+1, 2m} &= V_2^T c_k(n+1, m) & b_{2n+1, 2m+1} &= V_2^T d_k(n+1, m) \end{aligned} \quad (6)$$

式中  $c_k(n, m), d_k(n, m), i=1, 2, 3$  分别表示子图  $C_k, D_k$  中第  $m$  列、中点在第  $n$  行的相应于向量  $V_1, V_2$  长度的列向量, ( $V_2$  的长度为偶数, 此时中点视为中心以下的第一个位置).

计算出所有的  $a_{n,m}, b_{n,m}, n=0, 1, \dots, 2N-1; m=0, 1, \dots, 2M-1$ , 列滤波即告完成. 由于  $V_1, V_2$  的长度之和等于  $H$  的长度, 列滤波的计算量约为  $4MNK$ .

2) 行滤波

记

$$\begin{aligned} A(n, m) &= H_1^T [X(n, m) + Y^1(n, m-1) + Y^2(n-1, m) + Y^3(n-1, m-1)] \\ &\quad + H_2^T [X(n, m) + Y^1(n, m-1) - Y^2(n-1, m) - Y^3(n-1, m-1)] \\ B(n, m) &= H_1^T [X(n, m) - Y^1(n, m-1) + Y^2(n-1, m) - Y^3(n-1, m-1)] \\ &\quad + H_2^T [X(n, m) - Y^1(n, m-1) - Y^2(n-1, m) + Y^3(n-1, m-1)] \end{aligned}$$

则

$$\begin{aligned} c_{k-1, n, m} &= 2\{A(n, m)H_1 + B(n, m)H_2\} \\ n &= 0, 1, \dots, 2N-1; m = 0, 1, \dots, 2M-1 \end{aligned} \quad (7)$$

$A(n, m), B(n, m)$  的系数可由  $a_{n,m}, b_{n,m}$  确定, 事实上

i) 进行  $H_1$  滤波的各行系数 ( $m=0, 1, \dots, 2M-1$ )

第  $2n$  行  $a_{2n, m} - b_{2n, m}$       第  $2n+1$  行  $a_{2n+1, m} + b_{2n+1, m}$

ii) 进行  $H_2$  滤波的各行系数 ( $m=0, 1, \dots, 2M-1$ )

第  $2n$  行  $a_{2n, m} - b_{2n, m}$  ( $m$  为偶) 或  $-a_{2n, m} + b_{2n, m}$  ( $m$  为奇)

第  $2n+1$  行  $a_{2n+1, m} + b_{2n+1, m}$  ( $m$  为偶) 或  $-(a_{2n+1, m} + b_{2n+1, m})$  ( $m$  为奇)

在程序实现时, 滤波器  $H_1, H_2$  由  $V_1, V_2$  替代, 其计算量大约为

$$2M(K/2 - 1)2N + 2M(K/2 + 1)2N = 4MNK$$

该算法实现式(1)总的计算量约为  $8MNK$ .

### 2.3 利用 $V_1, V_2$ 对称性进一步减少滤波卷积运算

根据滤波器性质, 容易得知, 向量  $V_1, V_2$  分别关于中心对称, 利用这一性质可进一步减少卷积运算量.

例如, 在计算  $V_1^T c_k(n, m)$  时, 上述算法的程序代码是

```
result = 0;
for(i = -L_v; i <= L_v; i++)
result += V_1[i + L_v] * c_k[n + i, m];
```

共有  $(2L_v + 1)$  次浮点乘法和加法运算.

利用  $V_1[-i] = V_1[i]$ , 程序代码改为

```
result = V_1[L_v] * c_k[n, m];
```

for( $i = -L_v; i < 0; i++$ )

result +=  $V_1[i + L_v] * (c_k[n + i, m] + c_k[n - i, m]);$

计算量减少为 $(L_v + 1)$ 次乘法运算和 $(2L_v + 1)$ 次加法运算。

### 3 结 论

作者选用 23 点 Lemarie 滤波器<sup>[1]</sup>, 对标准的 8 bit/pixel 256x256 man 图象和 8 bit/pixel 512x512 baboon 图象分别进行了比较计算, 表 1 是本文方法和通用的 Mallat 重构算法的时间性能和图象质量比较结果。计算机仿真结果表明, 本方法有效地提高了图象重构速度, 并且重构图象客观质量有所提高。

表 1 图象经三级分解后重构的时间花费和信噪比

图 象	时间花费 (s)		PSNR(dB)	
	Mallat 算法	本文方法	Mallat 算法	本文方法
Man	11.92	6.15	37.39	40.35
Baboon	50.92	25.76	29.57	37.30

### 参 考 文 献

- 1 Mallat S. A theory for multiresolution signal decomposition; the wavelet representation. IEEE Trans Pattern Analysis and Machine Intelligence. 1989, 11(7):674~693
- 2 Antonini M. et al. Image coding using wavelet transform. IEEE Trans Image Processing. 1993, 1(2):205~220
- 3 Shapiro J. Embedded image coding using zerotrees of wavelet coefficients. IEEE Trans Signal Processing. 1993, 41(12):3445~3462
- 4 Mallat S. Multifrequency channel decompositions of images and wavelet models. IEEE Trans Acoustics, Speech and Signal Processing. 1989, 37(12):2091~2110

## An Image Reconstruction Fast Algorithm Based on Orthogonal Wavelets Derived from Symmetric Scale Functions

*Liu Yixu    Zhang Bangji    Wang Guangzai*

(Department of Automation, Chongqing University)

**ABSTRACT** A new image reconstruction algorithm is presented, based on orthogonal wavelets derived from symmetric scale functions. The analysis and practice show that the method lessens greatly the computational time, and improves the reconstruction quality.

**KEYWORDS** image transforms; filter; algorithms / orthogonal wavelets