

# 并行格型结构实现基于 DCT 的 2D 实值 离散 Gabor 变换

崔蓓蓓<sup>1</sup>, 陶 亮<sup>2</sup>

(1. 安徽大学 计算智能与信号处理实验室, 安徽 合肥 230039;

2. 安徽大学 计算机科学与技术学院, 安徽 合肥 230039)

**摘 要:**虽然 2D Gabor 变换在图像处理等很多领域认为是非常有用的时频分析的方法, 然而实时应用却因其很高的计算复杂性而受到限制。文中回顾了基于 DCT 的 2D 的实值离散 Gabor 变换, 为了有效地和快速地计算实值离散 Gabor 变换, 提出了在临界抽样条件下, 二维实值 Gabor 变换系数求解的块时间递归算法以及由变换系数重建原信号的块时间递归算法, 研究了该算法使用并行格型结构的实现方法, 并讨论和比较了算法的计算复杂性和优越性, 证明了基于 DCT 的 2D 实值离散 Gabor 变换块时间递归算法并行格型结构在计算复杂性的高性能。

**关键词:**基于 DCT 的 2D 的实值离散 Gabor 变换; 并行格型结构; 块时间递归算法

**中图分类号:** TP301.6

**文献标识码:** A

**文章编号:** 1673-629X(2009)02-0105-04

## Parallel Lattice Structure for Block Time - Recursive Algorithms of 2D Discrete Gabor Transforms Based on DCT

CUI Bei-bei<sup>1</sup>, TAO Liang<sup>2</sup>

(1. Computational Intelligence and Signal Processing Laboratory, Anhui University, Hefei 230039, China;

2. Department of Computer Science & Technology, Anhui University, Hefei 230039, China)

**Abstract:** 2D Gabor transforms are considered as an effective time - frequency analysis technique in diverse areas as image processing ; however, its real time applications have been limited due to its high computational complexity. In this paper, firstly the 2D real - valued discrete Gabor transform based on DCT will be reviewed; secondly, block - recursive algorithms for the efficient and fast computation of 2 - D RDGT coefficients and for the fast reconstruction of the original signal from the RDGT coefficients will be developed in critical sampling case ; thirdly, unified parallel lattice structures for the implementation of the algorithms will be studied; and finally, the computational complexity and the advantages of the proposed algorithms will be discussed and compared, and also proved its high performance of computation complexity.

**Key words:** 2D real - valued discrete Gabor transforms based on DCT; parallel lattice structures; block time recursive algorithms

## 0 引 言

二维的 Gabor 展开和变换被公认为图像处理的最 好的方法之一, 但计算量很大, 实时应用受到很大的限制, Gabor 于 1946 年在文献[1]中已提出了在连续余弦变换基础上引入了实值 Gabor 变换, 1995 年文献[2]又将其离散化, 但文献[2]的作者的研究仅局限于

临界的一维离散 Gabor 变换系数的求解, 文中提出了基于 DCT 的二维快速并行算法, 讨论了 2 - D RDGT 的块时间递归特性以及如何运用双层并行格结构快速实现二维的 RDGT 块时间递归算法的方法(临界抽样)。并可利用快速 2DCT, 2IDCT 算法来加速变换, 因此比传统的复值离散 Gabor 变换在计算、实现方面更为简单。

## 1 基于 DCT 的二维实值离散 Gabor 变换的定义

设一图像  $I(x, y)$ ,  $x = 0, 1, \dots, X - 1$ ;  $y = 0, 1, \dots, Y - 1$  被分成  $K \times L$  个维数为  $M \times N$  的不重叠的格子(lattice), 使得  $X = KM$  和  $Y = LN$ (临界抽样)。

收稿日期: 2008-06-04

基金项目: 国家自然科学基金(60572128); 安徽省人才开发基金(20052029)

作者简介: 崔蓓蓓(1983-), 女, 安徽亳州人, 硕士研究生, 研究方向为图像处理、多维信号; 陶 亮, 教授, 硕士生导师, 博士生导师, 研究方向为多维信号、模式识别与信息处理。

这样图像  $I(x, y)$  的实值离散 Gabor 展开就可定义为下列形式:

$$I(x, y) = \sum_{K=0}^{K-1} \sum_{L=0}^{L-1} \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} a(k, l, m, n) g_{klmn}(x, y) \quad (1)$$

系数  $a(k, l, m, n)$  可以通过下式获得:

$$a(k, l, m, n) = \sum_{x=0}^{X-1} \sum_{y=0}^{Y-1} I(x, y) \gamma_{klmn}(x, y) \quad (2)$$

其中

$$g_{klmn}(x, y) = \tilde{g}(x - kM, y - lN) \cdot \sqrt{\frac{2}{M}} g_m \cos\left(\frac{\pi(2\text{mod}(x, M) + 1)\text{mod}(m, M)}{2M}\right) \cdot \sqrt{\frac{2}{N}} g_n \cos\left(\frac{\pi(2\text{mod}(y, N) + 1)\text{mod}(n, N)}{2N}\right) \quad (3)$$

$$\gamma_{klmn}(x, y) = \tilde{\gamma}(x - kM, y - lN) \cdot \sqrt{\frac{2}{M}} g_m \cos\left(\frac{\pi(2\text{mod}(x, M) + 1)\text{mod}(m, M)}{2M}\right) \cdot \sqrt{\frac{2}{N}} g_n \cos\left(\frac{\pi(2\text{mod}(y, N) + 1)\text{mod}(n, N)}{2N}\right) \quad (4)$$

$$g_n = \begin{cases} 1/\sqrt{2}, n = 0 \\ 1, n \neq 0 \end{cases} \quad g_m = \begin{cases} 1/\sqrt{2}, m = 0 \\ 1, m \neq 0 \end{cases}$$

## 2 临界抽样条件下 2D-RDGT 的块时间递归算法

将图像  $I(x, y)$  以及  $a(k, l, m, n)$ 、 $\gamma(x, y)$  分别写成由  $K \times L$  个维数为  $M \times N$  的不重叠的块矩阵构成形式,即

$$\mathbf{I} = [\mathbf{I}_{kl}]_{k,l=0}^{K-1,L-1} = [\hat{\mathbf{I}}_k]_{k=0}^{K-1} \quad \mathbf{A} = [\mathbf{A}_{kl}]_{k,l=0}^{K-1,L-1}, \mathbf{\Gamma} = [\mathbf{\Gamma}_{kl}]_{k,l=0}^{K-1,L-1} \quad (5)$$

式中

$$\mathbf{I}_{kl} = [I(kM + i, lN + j)]_{i,j=0}^{M-1,N-1} \\ \mathbf{A}_{kl} = [a(k, l, m, n)]_{m,n=0}^{M-1,N-1} \\ \mathbf{\Gamma}_{kl} = [\gamma(kM + i, lN + j)]_{i,j=0}^{M-1,N-1} \\ \hat{\mathbf{I}}_k = [I(kM + i, j)]_{i,j=0}^{M-1,Y-1}$$

注意,  $\mathbf{I}$ 、 $\mathbf{A}$ 、 $\mathbf{\Gamma}$  可看成具有周期性的二维信号。

$$\hat{\mathbf{A}}_{kl} = \begin{bmatrix} \mathbf{\Gamma}_{(K-k)(L-l)} & \mathbf{\Gamma}_{(K-k)(L-l+1)} & \cdots & \mathbf{\Gamma}_{(K-k)0} & \cdots & \mathbf{\Gamma}_{(K-k)(L-l-1)} \\ \mathbf{\Gamma}_{(K-k+1)(L-l)} & \mathbf{\Gamma}_{(K-k+1)(L-l+1)} & \cdots & \mathbf{\Gamma}_{(K-k+1)0} & \cdots & \mathbf{\Gamma}_{(K-k+1)(L-l-1)} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \mathbf{\Gamma}_{(K-1)(L-l)} & \mathbf{\Gamma}_{(K-1)(L-l+1)} & \cdots & \mathbf{\Gamma}_{(K-1)0} & \cdots & \mathbf{\Gamma}_{(K-1)(L-l-1)} \\ \mathbf{\Gamma}_{0(L-l)} & \mathbf{\Gamma}_{0(L-l+1)} & \cdots & \mathbf{\Gamma}_{00} & \cdots & \mathbf{\Gamma}_{0(L-l-1)} \\ \mathbf{\Gamma}_{1(L-l)} & \mathbf{\Gamma}_{1(L-l+1)} & \cdots & \mathbf{\Gamma}_{10} & \cdots & \mathbf{\Gamma}_{1(L-l-1)} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \mathbf{\Gamma}_{(K-k-1)(L-l)} & \mathbf{\Gamma}_{(K-k-1)(L-l+1)} & \cdots & \mathbf{\Gamma}_{(K-k-1)0} & \cdots & \mathbf{\Gamma}_{(K-k-1)(L-l-1)} \end{bmatrix} * \begin{bmatrix} \hat{\mathbf{I}}_0 \\ \hat{\mathbf{I}}_1 \\ \vdots \\ \hat{\mathbf{I}}_{K-1} \end{bmatrix} \quad (8)$$

设两矩阵  $\mathbf{S} = [\mathbf{S}_{ij}]_{i,j=0}^{r-1,p-1}$ ,  $\mathbf{T} = [\mathbf{T}_{ij}]_{i,j=0}^{r-1,p-1}$ , 其中  $r, p$  为任意正整数, 子矩阵块  $\mathbf{S}_{ij}$  与  $\mathbf{T}_{ij}$  必须具有相同的矩阵维数  $M \times N$ 。定义两矩阵运算  $\mathbf{S} * \mathbf{T}$  为:

$$\mathbf{S} * \mathbf{T} = \sum_{i=0}^{r-1} \sum_{j=0}^{p-1} \mathbf{S}_{ij} \circ \mathbf{T}_{ij} = \sum_{i=0}^{r-1} \sum_{j=0}^{p-1} \mathbf{T}_{ij} \circ \mathbf{S}_{ij} \quad (6)$$

式中: “ $\circ$ ” 表示矩阵的 Hadamard 积<sup>[3]</sup>。

令  $x = \bar{k}M + \bar{m}$ ,  $y = \bar{l}N + \bar{n}$ , 利用式(4), 将式(2)改写成下式:

$$a(k, l, m, n) = \sum_{\bar{m}=0}^{M-1} \sum_{\bar{n}=0}^{N-1} \left[ \sum_{\bar{k}=0}^{K-1} \sum_{\bar{l}=0}^{L-1} \mathbf{I}(\bar{k}M + \bar{m}, \bar{l}N + \bar{n}) \cdot \gamma(\bar{k}M + \bar{m} - kM, \bar{l}N + \bar{n} - lN) \right] \cdot \sqrt{\frac{2}{M}} \cos\left(\frac{\pi(2\bar{m} + 1) \cdot m}{2M}\right) g_m \cdot \sqrt{\frac{2}{N}} \cos\left(\frac{\pi(2\bar{n} + 1) \cdot n}{2N}\right) g_n \quad (7)$$

由上式可看出任意一个 RDGT 系数块  $\mathbf{A}_{kl}$  为  $\hat{\mathbf{A}}_{kl}$  的  $M \times N$  点二维离散 DCT 变换 (2-D DCT), 即  $\mathbf{A}_{kl} = \text{DCT}(\hat{\mathbf{A}}_{kl})$ , 式中  $k = 0, 1, \dots, K-1$ ,  $l = 0, 1, \dots, L-1$ , 并且有式(8)。

设图像  $\mathbf{I}$  的一个块所在的行称为块行, 一个块行由  $L$  个块组成, 图像  $\mathbf{I}$  有  $K$  个块行, 即  $\hat{\mathbf{I}}_0, \hat{\mathbf{I}}_1, \dots, \hat{\mathbf{I}}_{K-1}$ , 如式(5)所示; 块行逐行自上而下被串行输入到用于 RDGT 系数计算的输入延时缓冲区 (如图 1 所示), 由式(8)定义块行时间  $t_r$  ( $t_r = 0, 1, 2, \dots$ ) 时刻的  $\hat{\mathbf{A}}_{kl}$  为下式, 以便导出上一块行时间与下一块行时间的 RDGT 系数计算的块行时间递归关系:

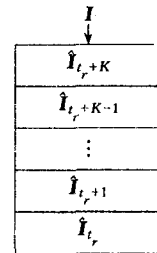


图 1 块行输入延时缓冲区

$$\hat{\mathbf{A}}_{kl}(t_r) = [\mathbf{\Gamma}_{(K-k)(L-l)} \mathbf{\Gamma}_{(K-k)(L-l+1)} \cdots \mathbf{\Gamma}_{(K-k)0} \cdots \mathbf{\Gamma}_{(K-k)(L-l-1)} \mathbf{\Gamma}_{(K-k)(L-l-1)}] * \hat{\mathbf{I}}_{t_r} + [\mathbf{\Gamma}_{(K-k+1)(L-l)} \mathbf{\Gamma}_{(K-k+1)(L-l+1)} \cdots \mathbf{\Gamma}_{(K-k+1)0} \cdots \mathbf{\Gamma}_{(K-k+1)(L-l-1)}]$$

$$\begin{aligned}
& \Gamma_{(K-k+1)0} \cdots \Gamma_{(K-k+1)(L-l-1)}] * \hat{I}_{t_r+1} + \cdots + \\
& [\Gamma_{(K-1)(L-l)} \Gamma_{(K-1)(L-l+1)} \cdots \Gamma_{(K-1)0} \cdots \Gamma_{(K-1)(L-l-1)}] * \\
& \hat{I}_{t_r+i-1} + [\Gamma_{0(L-l)} \Gamma_{0(L-l+1)} \cdots \Gamma_{00} \cdots \Gamma_{0(L-l-1)}] * \hat{I}_{t_r+i} + \\
& [\Gamma_{1(L-l)} \Gamma_{1(L-l+1)} \cdots \Gamma_{10} \cdots \Gamma_{1(L-l-1)}] * \hat{I}_{t_r+i+1} + \cdots + \\
& [\Gamma_{(K-k-1)(L-l)} \Gamma_{(K-k-1)(L-l+1)} \cdots \Gamma_{(K-k-1)0} \cdots \\
& \Gamma_{(K-k-1)(L-l-1)}] * \hat{I}_{t_r+K-1} \quad (9)
\end{aligned}$$

由上式不难导出在块行时间  $t_r + 1$  时刻的  $\hat{A}_{kl}$  为

$$\begin{aligned}
\hat{A}_{kl}(t_r + 1) &= \hat{A}_{(k+1)l}(t_r) + [\Gamma_{(K-k-1)(L-l)} \\
&\Gamma_{(K-k-1)(L-l+1)} \cdots \Gamma_{(K-k-1)0} \cdots \Gamma_{(K-k-1)(L-l-1)}] * \\
&(\hat{I}_{t_r+K} - \hat{I}_{t_r}) \quad (10)
\end{aligned}$$

$K$  个块行图像全部进入输入缓冲区之后,即在块行时间  $t_r > K - 1$  时,即式(10)中  $\hat{I}_{t_r+K} = \hat{I}_{t_r}$ ,于是式(10)的递归过程进入稳定阶段。因此可在块行时间  $t_r = K$  时刻,由式(10)解出图像的 RDGT 系数。

在上述递归过程中,实际上在块行时间  $t_r \leq K - 1$  时,缓冲区中  $\hat{I}_{t_r} = 0$ 。令当前  $t_r$  时刻缓冲区中  $\hat{I}_{t_r+K}$  均分为  $L$  个子块,即  $\hat{I}_{t_r+K} = [\bar{I}_0 \bar{I}_1 \cdots \bar{I}_{L-1}]$ ,并在块行时间  $0 \leq t_r \leq K - 1$ ,令式(10)右边最后一项为

$$\begin{aligned}
\bar{A}_{kl} &= [\Gamma_{(K-k-1)(L-l)} \Gamma_{(K-k-1)(L-l+1)} \cdots \Gamma_{(K-k-1)(L-l)} \\
&\Gamma_{(K-k-1)0} \cdots \Gamma_{(K-k-1)(L-l-1)}] * [\bar{I}_0 \bar{I}_1 \cdots \bar{I}_{L-1}] \quad (11)
\end{aligned}$$

设块行图像按横向逐块自左向右输入到块延时缓冲区(如图2所示),定义块与块之间的输入时间间隔为一块时间间隔,因此一块行时间间隔等于  $L$  个块时间间隔。

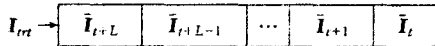


图2 块输入延时缓冲区

由式(11)定义块时间  $t$  ( $t = 0, 1, 2, \dots$ ) 时刻的  $\bar{A}_{kl}$  为:

$$\begin{aligned}
\bar{A}_{kl}(t) &= [\Gamma_{(K-k-1)(L-l)} \Gamma_{(K-k-1)(L-l+1)} \cdots \\
&\Gamma_{(K-k-1)(L-l)} \Gamma_{(K-k-1)0} \cdots \Gamma_{(K-k-1)(L-l-1)}] * \\
&[\bar{I}_{t+1} \cdots \bar{I}_{t+L-1}] \quad (12)
\end{aligned}$$

因此,块时间  $t + 1$  时刻的  $\bar{A}_{kl}$  为

$$\begin{aligned}
\bar{A}_{kl}(t + 1) &= \bar{A}_{k(l+1)}(t) + \Gamma_{(K-k-1)(L-l-1)} \cdot \\
&(\bar{I}_{t+L} - \bar{I}_t) \quad (13)
\end{aligned}$$

式(13)中  $\bar{I}_{t+L} = \bar{I}_t$ ,递归过程进入稳定阶段。但在实际运算中,为了使得计算  $\bar{A}_{kl}$  与  $\hat{A}_{kl}$  的递归过程在时间上衔接起来,取块时间  $t = L$  时刻,将式(13)的递归解  $\bar{A}_{kl}$  送入到块行递归过程用于  $\hat{A}_{kl}$  的计算。注意  $t \leq L - 1$  时,式(13)中  $\bar{I}_t = 0$ 。

综合上述,求解 RDGT 系数块  $A_{kl}$  ( $k = 0, 1, \dots, K - 1, l = 0, 1, \dots, L - 1$ ) 算法步骤如下:

(1) 令  $t_r = 0, t = 0$ ,将所有  $\bar{A}_{kl}, \hat{A}_{kl}$  ( $k = 0, 1, \dots,$

$K - 1, l = 0, 1, \dots, L - 1$ ) 清零。

(2) 输入第  $t_r$  块行中的第  $t$  块图像  $I_{t_r}$ ,即在输入缓冲区中  $\bar{I}_{t+L} = I_{t_r}$ ,对所有  $k, l$  计算:

$$\bar{A}_{kl}(t + 1) = \bar{A}_{k(l+1)}(t) + \Gamma_{(K-k-1)(L-l-1)} \cdot \bar{I}_{t+L}$$

若  $t = L - 1$ ,将所有  $\bar{A}_{kl}$  代入步骤(3),然后将所有  $\bar{A}_{kl}$  清零,否则令  $t \leftarrow t + 1$ ,重复步骤(2)。

(3) 对所有  $k, l$  计算:

$$\hat{A}_{kl}(t_r + 1) = \hat{A}_{(k+1)l}(t_r) + \bar{A}_{kl}$$

若  $t_r = K - 1$ ,将所有  $\hat{A}_{kl}$  代入步骤(4),否则令  $t_r \leftarrow t_r + 1, t \leftarrow 0$ ,重复上述步骤(2)。

(4) 对所有  $k, l$  计算:

$$A_{kl} = \text{DCT}(\hat{A}_{kl})$$

### 3 由2-D RDGT 系数重建图像的块时间递归算法

首先将  $g(x, y)$  写成由  $K \times L$  个维数为  $M \times N$  的不重叠的块矩阵构成形式

$$G = [G_{kl}]_{k,l=0}^{K-1,L-1} \quad (14)$$

其中

$$G_{kl} = [g(-kM + i, -lN + j)]_{i,j=0}^{M-1,N-1}$$

令  $x = kM + m, y = lN + n, k = 0, 1, \dots, K - 1, l = 0, 1, \dots, L - 1, m = 0, 1, \dots, M - 1, n = 0, 1, \dots, N - 1$ ,利用式(2)将式(1)改写成下式

$$\begin{aligned}
I(kM + m, lN + n) &= \sum_{k=0}^{K-1} \sum_{l=0}^{L-1} g(kM + m - \bar{k}M, \\
&lN + n - \bar{l}N) \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} a(\bar{k}, \bar{l}, \bar{m}, \bar{n}) \sqrt{\frac{2}{M}} \cos \\
&(\frac{\pi(2m+1)\bar{m}}{2M}) g_m \cdot \sqrt{\frac{2}{N}} \cos(\frac{\pi(2n+1)\bar{n}}{2N}) g_n \quad (15)
\end{aligned}$$

显然,上式中后两个求和项为  $\bar{A}_{kl}$  的2-D DCT,结构和式(7)类同,同样可以写成矩阵形式进行块递归运算。

### 4 双层并行格型结构实现2-D RDGT 块时间递归算法

根据求解 RDGT 系数递归算法以及重建图像  $I$  递归算法设计的双层并行格型结构分别如图3和如图4所示。由图可见,每层并行格型结构中的并行处理单元结构都是一致的,并且十分紧凑,另外,DCT的实现比DFT要简单得多<sup>[4]</sup>,因此双层并行格型结构非常有利于软件与硬件的实现。

在图3中,对于每一个并行输出的 RDGT 系数块  $A_{kl}$ ,其计算复杂性为  $M \times N$  点2-D DCT +  $K \times L \times$

$M \times N$  次实数乘。

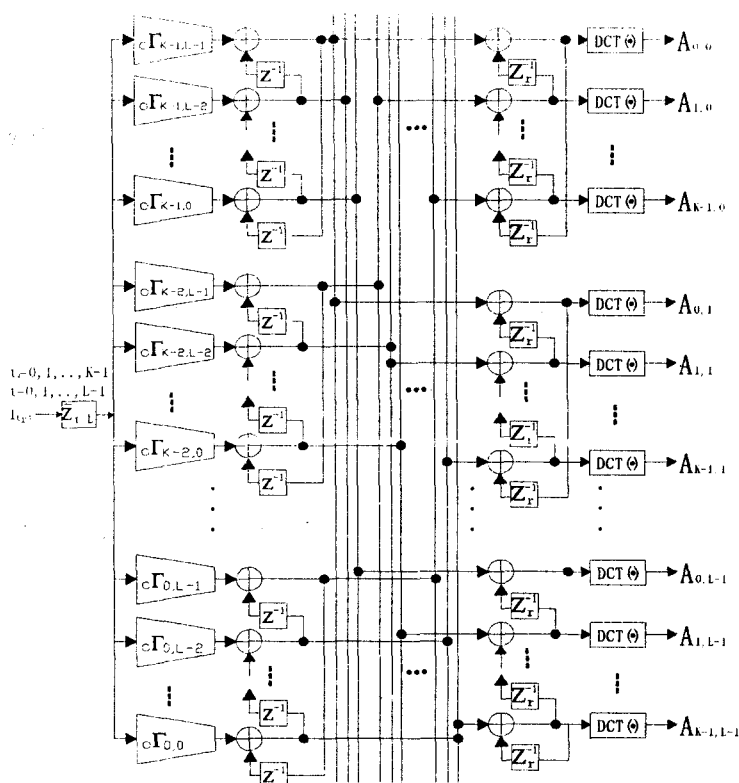


图 3 双层并行格型结构实现 2-D RDGT 系数求解

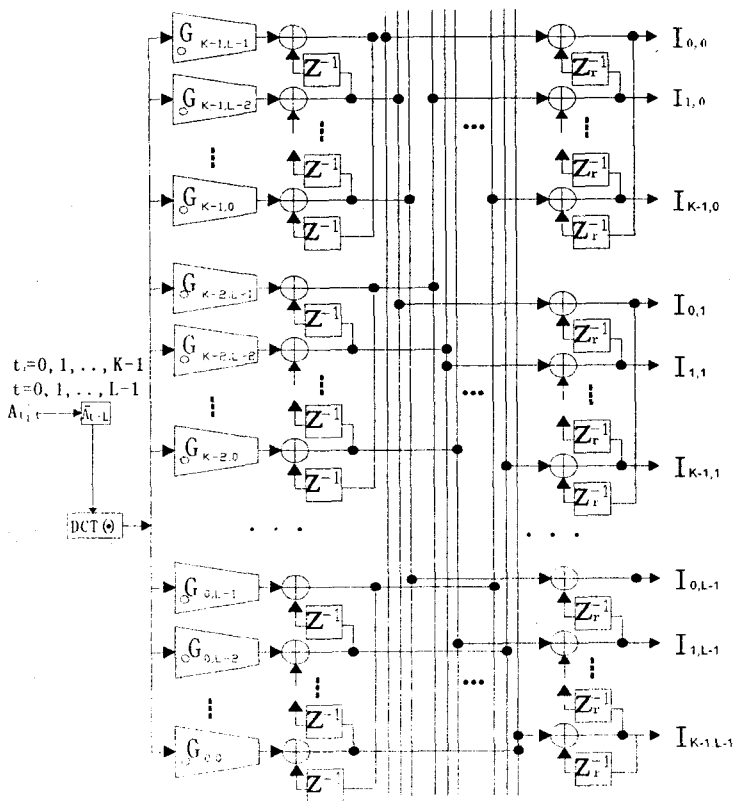


图 4 双层并行格型结构实现由 2-D RDGT 系数重建原图

在图 4 中,对于每一个并行输出的重建图像块  $I_{kl}$ ,

其计算复杂性为  $K \times L \times (M \times N \text{ 点 } 2-D \text{ DCT}) + K \times L \times M \times N \text{ 次实数乘}$ , 计算复杂性明显小于目前二维复值离散 Gabor 变换串行算法的计算复杂性, 如文献[5]中二维复值离散 Gabor 变换系数及图像重建的计算复杂性皆为  $K \times L \times (M \times N \text{ 点 } 2-D \text{ DFT}) + K \times L \times X \times Y \text{ 次复数相乘}$ , 其中  $X = K \times M, Y = L \times N$  (临界抽样)。而在文献[6]和[7]中计算复杂性为  $K \times L \times M \times N \text{ 点 } 4-D \text{ DFT} + K \times L \times M \times N \text{ 点 } 4-D \text{ IDFT} + X \times Y \text{ 次复数除} + X^2 \times Y^2 \text{ 次复数乘}$ , 并且这些算法均涉及复数运算和 FFT, 以上计算复杂性分析说明了基于 DCT 的 RDGT 时间递归算法的并行格型结构在计算时间方面所具有的高速和高效性能。

## 5 结束语

提出了基于 DCT 的 2D 实值离散 Gabor 变换并行格型结构实现方法(临界抽样), 这种方法只涉及实数运算, 并可借助快速余弦算法实现。验证了并行格型结构实现的可行性, 计算复杂性分析与比较说明了 DCT 的 2D 实值离散 Gabor 变换块时间递归算法并行格结构在计算时间方面的高速性。

## 参考文献:

- [1] Gabor D. Theory of communication[J]. J Inst Electr Eng, 1946, 9(3): 429-457.
- [2] Stewart D F, Potter L C, Ahalt S C. Computationally Attractive Real Gabor Transforms[J]. IEEE Transactions on Signal Processing, 1995, 43(1): 77-84.
- [3] 张贤达. 非平稳信号分析与处理[M]. 北京: 科学出版社, 1997.
- [4] Bracewell R N. The Fourier Transform and Its Application[M]. 2nd ed. New York: McGraw-Hill, 1986.
- [5] Wang L, Chen C T, Lin W C. An efficient algorithm to compute the complete set of discrete Gabor coefficients[J]. IEEE Transactions on Image Processing, 1994, 3(1): 87-92.
- [6] Baxter R A. SAR Image compression with the Gabor Transform[J]. IEEE Transactions on Geoscience and Remote Sensing, 1999, 37(1): 574-588.
- [7] Bracewell R N. The Discrete Hartley Transform[J]. J. Opt. Soc. Am., 1983, 73(12): 1832-1835.