权重化QR分解的正交匹配追踪算法硬件实现

王 玺,梁文凯,杨 虹,张红升,刘 挺,牟晓霜,张 磊,余柏汕,黎 淼*
(重庆邮电大学光电工程学院,重庆 400065)

摘 要: 为在小型化、低成本的硬件平台实现正交匹配追踪(Orthogonal Matching Pursuit,OMP)算法,针对OMP算法中最小二乘法的问题,该文构造一个确定性的传感矩阵,提出一种低复杂度、低资源的权重化QR分解的OMP(Weighted QR decomposition OMP,WQR-OMP)算法硬件结构,在ZYNQ 7020型号芯片上搭建WQR-OMP SOC系统.WQR-OMP算法在传感矩阵进行QR分解后,根据三角矩阵$\boldsymbol{R}$中元素的分布特性,通过权重化运算只保留主对角线上的元素而其他余元素归零,得到对角矩阵$\boldsymbol{D}$,然后近似计算稀疏向量的解.实验结果表明:与基于QR分解的OMP(QR decomposition OMP,QR-OMP)和Batch-OMP算法的硬件结构相比,WQR-OMP算法硬件结构的重构速度更快、存储资源更少.在压缩率为0.25的条件下,WQR-OMP SOC系统对256×256分辨率图像的重构时间为400 ms左右,其速率比仅使用ARM处理器的重构速率提高了约6.3倍.与其他现有研究对比,该系统在Block RAM存储资源消耗较少的情况下,进一步提升了重构速度,适用于存储资源受限的硬件平台.

关键词: 正交匹配追踪算法;最小二乘;权重化;QR分解;ZYNQ 7020

基金项目: 国家自然科学基金(No.61604028);重庆市技术创新与应用发展专项重点项目(No.cstc2020jscx-gksbX0012);重庆市基础研究与前言探索重点项目(No.cstc2021ycjh-bgzxm0085)

中图分类号: TN911.7 **文献标识码:** A **文章编号:** 0372-2112(2024)05-1534-09

电子学报URL:http://www.ejournal.org.cn **DOI:**10.12263/DZXB.20220554

Hardware Implementation of Orthogonal Matching Pursuit Algorithm for Weighted QR Decomposition

WANG Xi, LIANG Wen-kai, YANG Hong, ZHANG Hong-sheng, LIU Ting, MOU Xiao-shuang, ZHANG Lei, YU Bai-shan, LI Miao*
(*School of Optoelectronic Engineering*, *Chongqing University of Posts and Telecommunications*, *Chongqing* 400065, *China*)

Abstract: To realize the orthogonal matching pursuit (OMP) algorithm on a miniaturized and low-cost hardware platform, for calculation of the least square method in the OMP algorithm, this paper constructs a deterministic perception matrix and proposes a low-complexity, low-resource weighted QR decomposition OMP (WQR-OMP) algorithm hardware architecture, and the WQR-OMP SOC system is built on the ZYNQ 7020 chip. The WQR-OMP algorithm is that after the QR decomposition of the sensing matrix according to the distribution characteristics of the elements in the triangular matrix $\boldsymbol{R}$, the elements on the main diagonal are retained through the weighting operation, which returns other elements to zero to obtain the diagonal matrix $\boldsymbol{D}$, and then approximately computes the solution for the sparse vector. The experimental results show that compared with the hardware architecture of OMP algorithm based on QR decomposition OMP (QR-OMP) and Batch-OMP algorithm, the WQR-OMP algorithm has lower computational complexity and fewer storage resources. The reconstruction time of the WQR-OMP SOC system is about 400 ms for 256×256 resolution images at a compression rate of 0.25, which is 6.3 times faster than the ARM processor does. Compared with other existing researchers, this system further improves the reconstruction speed with less consumption of Block RAM storage resources and is suitable for hardware platforms with limited storage resources.

Key words: orthogonal matching pursuit algorithm; least squares; weighted; QR decomposition; ZYNQ 7020

收稿日期:2022-05-14;修回日期:2023-12-28;责任编辑:孙瑶
*通讯作者:黎淼

Foundation Item(s): National Natural Science Foundation of China (No.61604028); Special Key Project of Chongqing Technology Innovation and Application Development (No.cstc2020jscx-gksbX0012); Key Projects of Basic Research and Preface Exploration in Chongqing (No.cstc2021ycjh-bgzxm0085)

1　引言

压缩感知(Compressive Sensing,CS)是一种新型的信号采样理论. 该理论在2006年[1,2]被提出,为信号处理领域开辟了新的探索方向. 该理论突破基于香农定理的信号采样方式,实现对原始信号的"边压缩,边采样". 其原理是通过将可稀疏表示的原始信号由高维空间投影到低维空间,在远低于Nyquist采样速率的情况下,对低维空间的信号进行随机采样,获得少量的测量信号,最终由重构算法恢复出原始信号. CS理论具有存储资源少和硬件成本低的优势,在无线通信、核磁共振和单像素成像(Single-Pixel Imaging,SPI)等[3~5]领域得到广泛应用.

在CS理论的信号重构过程中,利用重构算法从少量测量信号恢复出原始信号涉及大量计算,信号恢复时间较长. 目前,大多数重构算法在计算机端上实现,存在功耗大、成本高和体积大等问题. 因此,本研究在小型化、低成本的硬件平台设计重构算法的硬件结构,提供一种便携式、低功耗的CS重构系统.

CS理论的重构算法主要包括贪婪算法、凸优化算法和非凸优化算法三大类[6]. 其中贪婪算法中的正交匹配追踪(Orthogonal Matching Pursuit,OMP)算法是目前硬件实现最常用的算法之一. OMP算法包括两个耗时的计算阶段:(1)寻找最大匹配原子,其中涉及矩阵与向量内积计算;(2)最小二乘法,其中涉及线性方程组求解稀疏向量的计算[7]. 针对最小二乘法计算,常采用矩阵分解的方法优化线性方程组求解的计算,达到降低计算复杂度的目的. 2015年,Rabah等人[8]提出一种求解线性系统的并行、低复杂度的体系结构,使用修正的Cholesky分解优化最小二乘法的计算,然后使用回代法求解稀疏向量,提高计算效率. 2019年,Ge等人[9]提出一种无平方根的QR分解,避免了平方根的计算,并在迭代完成后使用回代法计算稀疏向量. 2021年Li等人[10]使用Gram-Schmidt方式改进残差的更新过程,每次迭代过程无需计算稀疏向量,待迭代完成后使用修正的Cholesky分解优化最小二乘法的计算,提高了OMP算法的计算效率.

在使用修正的Cholesky分解优化最小二乘法的计算时,虽然线性方程组求解的复杂度降低了,但每次迭代过程仍需使用回代法计算稀疏向量. 而使用QR分解可以优化更新残差的计算,避免更新残差计算对稀疏向量的依赖,在迭代完成后只需使用一次回代法计算就可以得到稀疏向量的解. 这样做的代价是增加了存储资源的消耗. 另外,目前大多数的研究只是针对OMP算法本身进行优化的,而未考虑输入的传感矩阵对OMP算法的影响. 为在资源受限的嵌入式硬件平台上实现快速计算,一般先计算出传感矩阵,并存储到内存单元中,避免复杂的传感矩阵计算.

为进一步降低计算复杂度和减少存储资源,本文构造一种特殊分布的传感矩阵,提出一种快速、低资源的权重化QR分解的OMP(Weighted QR decomposition OMP,WQR-OMP)算法,使用Xilinx Vivado HLS开发工具设计WQR-OMP算法的硬件结构,最终在ZYNQ 7020型号芯片上实现WQR-OMP SOC系统. 实验结果表明,与基于QR分解的OMP(QR decomposition OMP,QR-OMP)算法和Batch-OMP[11]算法相比,WQR-OMP算法硬件结构的重构速度较快且资源消耗较少. 在压缩率为0.25的条件下,WQR-OMP SOC系统能够实现分辨率为256×256的图像快速重构,其速率是ARM处理器重构速率的6.3倍. 与其他工作相比,该系统在低成本的硬件平台使用了较少的Block RAM存储资源,具有资源消耗较少、重构速度快的优势.

2　基于QR分解的OMP算法

CS理论的数学模型如图1所示. 假设原始信号x为一维$N\times 1$的信号,测量矩阵$\boldsymbol{\Phi}$的维度为$M\times N$,则信号x的压缩测量过程可表示为

$$y=\boldsymbol{\Phi}x \tag{1}$$

其中,$M<N$,y是维度为$M\times 1$的测量值. 为求解信号x,需要保证x是稀疏信号. 然而现实生活中大多数的信号本身并不是稀疏的,但可以在某些稀疏基上具有稀疏性,则信号x的稀疏表示为

$$x=\boldsymbol{\Psi}\boldsymbol{\theta} \tag{2}$$

其中,稀疏基$\boldsymbol{\Psi}$是维度为$N\times N$的正交矩阵;$\boldsymbol{\theta}$为$N\times 1$的稀疏向量,是信号x在稀疏基$\boldsymbol{\Psi}$的稀疏表示. 如果$\boldsymbol{\theta}$中有$K(K\ll N)$个非零元素(或较大元素),而其他元素为零(或较小元素),则稀疏向量$\boldsymbol{\theta}$是K稀疏的[12,13],即信号x在稀疏基$\boldsymbol{\Psi}$中以K稀疏的$\boldsymbol{\theta}$信号等价表示. 常见的稀疏基有离散余弦变换(Discrete Cosine Transform,DCT)、离散小波变换(Discrete Wavelet Transform,DWT)和离散傅里叶变换(Discrete Fourier Transform,DFT)[14]. 由式(1)和式(2),可得

$$y=\boldsymbol{\Phi}x=\boldsymbol{\Phi}\boldsymbol{\Psi}\boldsymbol{\theta}=\boldsymbol{A}\boldsymbol{\theta} \tag{3}$$

其中，$\boldsymbol{A}=\boldsymbol{\Phi\Psi}$是传感矩阵，维度为$M\times N$. 为计算稀疏向量$\boldsymbol{\theta}$，传感矩阵需要满足有限等距性质（Restricted Isometry Property，RIP）[15]条件，即

$$(1-\delta)\|\boldsymbol{\theta}\|_2^2\leqslant\|\boldsymbol{A\theta}\|_2^2\leqslant(1+\delta)\|\boldsymbol{\theta}\|_2^2 \tag{4}$$

其中，$\delta\in(0,1)$. 在实际应用中验证传感矩阵$\boldsymbol{A}$是否满足RIP条件往往比较困难，可以使用传感矩阵$\boldsymbol{A}$的相关性作为判定的条件[16~18]，即

$$\mu(\boldsymbol{A})=\max_{1\leqslant i<j\leqslant N}\frac{\left|\left\langle\boldsymbol{a}_i,\boldsymbol{a}_j\right\rangle\right|}{\|\boldsymbol{a}_i\|_2\|\boldsymbol{a}_j\|_2} \tag{5}$$

其中，$\boldsymbol{a}_i$和$\boldsymbol{a}_j$分别表示传感矩阵$\boldsymbol{A}$中的列向量；$\mu(\boldsymbol{A})\in\left[\sqrt{\frac{N-M}{M(N-1)}},1\right]$表示相干系数. 在式(3)中稀疏向量$\boldsymbol{\theta}$重构是利用重构算法从低维空间恢复到高维空间的过程，即求解一个欠定方程组$\boldsymbol{y=A\theta}$的问题. 由于该方程组有无穷多个解，因此信号重构过程可转化为求解最优化问题，可以表示为

$$\min\|\hat{\boldsymbol{\theta}}\|_0\quad\text{s.t.}\quad\boldsymbol{y=A}\hat{\boldsymbol{\theta}} \tag{6}$$

通过重构算法求出稀疏向量近似解$\hat{\boldsymbol{\theta}}$，再由式(2)中信号稀疏表示$\hat{x}=\boldsymbol{\Psi}\hat{\boldsymbol{\theta}}$，求得原始信号的近似解$\hat{x}$.

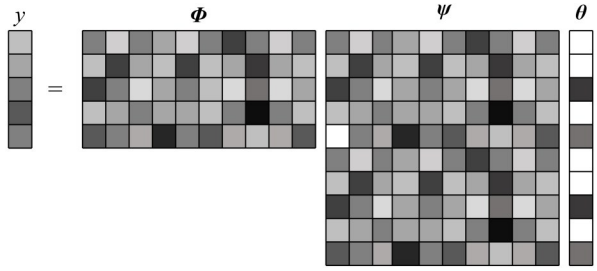


图1 CS理论的数学模型

针对式(6)的最优化问题，本文使用OMP算法进行求解. OMP算法计算流程如算法1所示. 首先残差r_{t-1}与传感矩阵$\boldsymbol{A}$做内积计算，选出内积绝对值最大的列，更新索引集Λ_t和原子集$\boldsymbol{A}_t$. 其次通过最小二乘法求解稀疏向量$\boldsymbol{\theta}_t$，然后更新残差r_t，执行迭代判断. 最终迭代结束后通过θ_K和Λ_K得到稀疏向量$\hat{\boldsymbol{\theta}}$，然后利用式(2)计算得到原始信号的近似值$\hat{x}$.

算法1中：t为当前迭代的次数；r_t为第t次迭代的残差；Λ_t为第t次迭代列序号λ_t的索引集；$\boldsymbol{A}_t$为第t次迭代的原子集；$\boldsymbol{\theta}_t$为第t次迭代的稀疏向量，维度为$t\times1$；$\hat{\boldsymbol{\theta}}$为最终的稀疏向量，维度为$N\times1$.

在算法1中，流程4和流程5的两个计算过程分别为

算法1 OMP

输入: 测量值$y\in R^{M\times1}$;传感矩阵$\boldsymbol{A=\Phi\Psi},\boldsymbol{A}\in\mathbf{R}^{M\times N}$;稀疏度$K$

输出: 重构信号$\hat{x}\in\mathbf{R}^{N\times1}$

算法流程:

1. 初始化:$t=1,r_0=y,\Lambda_0=\varnothing,\boldsymbol{A}_0=\varnothing$
2. 寻找最大匹配原子列的索引号:$\lambda_t=\arg\max_{1\leqslant j\leqslant N}\left|\left\langle r_{t-1},\boldsymbol{a}_j\right\rangle\right|$
3. 更新索引和原子集:$\Lambda_t=\Lambda_{t-1}\cup\{\lambda_t\};\boldsymbol{A}_t=\boldsymbol{A}_{t-1}\cup\{a_{\lambda_t}\}$
4. 最小二乘法:$y=\boldsymbol{A}_t\boldsymbol{\theta}_t$,即$\boldsymbol{\theta}_t=\arg\min_{\theta_t}\|y-\boldsymbol{A}_t\boldsymbol{\theta}_t\|=\left(\boldsymbol{A}_t^{\mathrm{T}}\boldsymbol{A}_t\right)^{-1}\boldsymbol{A}_t^{\mathrm{T}}y$
5. 更新残差:$r_t=y-\boldsymbol{A}_t\boldsymbol{\theta}_t$
6. 迭代判断:$t=t+1$,若$t\leqslant K$,执行流程2,否则执行流程7
7. 重构信号:把重构所得$\boldsymbol{\theta}_K$的值放入$\hat{\boldsymbol{\theta}}\left(\hat{\boldsymbol{\theta}}\in\mathbf{R}^{N\times1}\right)$中对应的$\Lambda_K$位置中,利用稀疏基求得重构信号,即$\hat{x}=\boldsymbol{\Psi}\hat{\boldsymbol{\theta}}$

$$\boldsymbol{\theta}_t=\left(\boldsymbol{A}_t^{\mathrm{T}}\boldsymbol{A}_t\right)^{-1}\boldsymbol{A}_t^{\mathrm{T}}y \tag{7}$$

$$r_t=y-\boldsymbol{A}_t\boldsymbol{\theta}_t \tag{8}$$

其中，残差r_t的计算依赖于稀疏向量$\boldsymbol{\theta}_t$，导致每次迭代过程需要计算$\boldsymbol{\theta}_t$. 稀疏向量$\boldsymbol{\theta}_t$涉及线性方程组求解运算，计算复杂度较高，因此需要消耗大量时间. 并且每次迭代过程$\boldsymbol{A}_t$的维度在不断改变，导致最小二乘法的计算维度也在改变，对于硬件实现较为困难. 于是，采用基于Gram－Schmidt方式的QR分解对算法1中的最小二乘法和更新残差两个计算过程进行优化[19]. 首先对原子集$\boldsymbol{A}_t$进行QR分解，可以表示为

$$\boldsymbol{A}_t=\boldsymbol{Q}_t\boldsymbol{R}_t \tag{9}$$

其中，$\boldsymbol{Q}_t$是维度为$M\times t$的标准正交矩阵；$\boldsymbol{R}_t$是维度为$t\times t$的上三角矩阵. 然后将式(9)带入式(7)中化简最小二乘法的计算，则稀疏向量$\boldsymbol{\theta}_t$的计算可表示为

$$\boldsymbol{\theta}_t=\boldsymbol{R}_t^{-1}\boldsymbol{Q}_t^{\mathrm{T}}y \tag{10}$$

最后将式(9)和式(10)带入更新残差的计算过程式(8)中，化简可得

$$r_t=y-\boldsymbol{Q}_t\boldsymbol{Q}_t^{\mathrm{T}}y \tag{11}$$

在式(11)中残差更新过程并未使用稀疏向量$\boldsymbol{\theta}_t$计算，则每次迭代过程可以避免式(10)中稀疏向量$\boldsymbol{\theta}_t$的计算. 在K次迭代完成后，只执行一次稀疏向量$\boldsymbol{\theta}_K$的计算，即

$$\boldsymbol{\theta}_K=\boldsymbol{R}^{-1}\boldsymbol{Q}_K^{\mathrm{T}}y \tag{12}$$

其中，$\boldsymbol{\theta}_K$为第K次迭代的稀疏向量，维度为$K\times1$. 此时，通过回代法即可求出稀疏向量$\boldsymbol{\theta}_K$. 虽然使用QR分解降低了OMP算法的计算复杂度，但是额外增加了矩阵$\boldsymbol{Q}$和矩阵$\boldsymbol{R}$的存储资源. 另外，在使用回代法计算稀疏向量$\boldsymbol{\theta}_K$时，由于三角矩阵$\boldsymbol{R}_K$的特性，$\boldsymbol{\theta}_K$中的每个元素存在依赖性，只能顺序执行.

3　WQR-OMP算法与硬件结构

在小型化、低成本的硬件平台实现OMP算法,除了需要对算法优化之外,还应对输入的传感矩阵$\boldsymbol{A}$进行研究. 为了便于在硬件上实现,一般会选用确定性的传感矩阵$\boldsymbol{A}$并提前存储到内存单元中,避免额外的计算操作. 另外,还可以根据传感矩阵$\boldsymbol{A}$元素的分布特性,通过舍去部分数据来减少计算复杂度和存储资源,进一步优化OMP算法的性能.

为解决QR分解存在的问题,我们希望传感矩阵$\boldsymbol{A}$进行QR分解后矩阵$\boldsymbol{R}$的元素分布如图2所示,即主对角线上的元素远大于其余元素. 然后通过权重化运算,保留主对角线元素,其余元素清零,进一步降低OMP算法的计算复杂度,减少存储资源的消耗. 因此,本文构造满足图2分布条件的传感矩阵$\boldsymbol{A}$,提出一种快速、低资源的WQR-OMP算法,使用Xilinx Vivado HLS开发工具设计相应的硬件结构,最终在ZYNQ 7020芯片上实现WQR-OMP SOC.

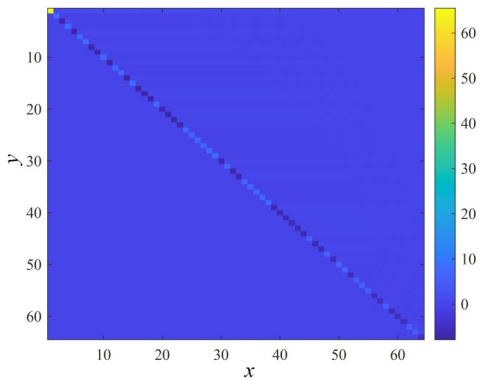


图2　矩阵$\boldsymbol{R}$中元素的亮度分布

3.1　WQR-OMP算法

本文使用基于Haar小波系数排序的Hadamard矩阵作为测量矩阵$\boldsymbol{\Phi}$[20],DCT矩阵作为稀疏基,构造一个确定性的传感矩阵$\boldsymbol{A}$. 然后对传感矩阵$\boldsymbol{A}=\boldsymbol{\Phi\Psi}$进行QR分解,矩阵$\boldsymbol{R}$中元素呈现出如图2所示的分布. 通过分析,$\boldsymbol{R}$中主对角线上的元素亮度变化比较明显,而上三角元素亮度值接近于0. 由于原子集$\boldsymbol{A}_K$是由残差r_t与传感矩阵$\boldsymbol{A}$经过K次迭代运算选择出的原子,因此原子集$\boldsymbol{A}_K$具有与传感矩阵$\boldsymbol{A}$相同的特性,即$\boldsymbol{A}_K$进行QR分解后的矩阵$\boldsymbol{R}_K$也呈现如图2所示的分布. 根据图2的分布特性,$\boldsymbol{R}_K$的权重化运算可表示为

$$\boldsymbol{D}=\boldsymbol{R}_K\circ\boldsymbol{W}_K \tag{13}$$

其中,$\boldsymbol{W}_K$是维度为$K\times K$的权重矩阵. 由于$\boldsymbol{R}_K$中上三角元素值接近于0,因此权重矩阵$\boldsymbol{W}_K$为单位矩阵;"$\circ$"是Hadamard乘积,表示矩阵对应元素相乘. 矩阵$\boldsymbol{D}$是维度为$K\times K$的对角矩阵. 将式(12)中的三角矩阵$\boldsymbol{R}_K$替换为矩阵$\boldsymbol{D}$,则稀疏向量$\boldsymbol{\theta}_K$可表示为

$$\boldsymbol{\theta}_K=\boldsymbol{D}^{-1}\boldsymbol{Q}_K^{\mathrm{T}}y \tag{14}$$

图3所示为改变前后稀疏向量$\boldsymbol{\theta}_K$的计算形式. 对比式(12)和式(14)稀疏向量$\boldsymbol{\theta}_K$的计算过程,在使用矩阵$\boldsymbol{R}_K$计算时,乘法和除法计算次数为$\frac{K(K+1)}{2}$,减法计算次数为$\frac{K(K-1)}{2}$. 而使用矩阵$\boldsymbol{D}$计算时,仅需要K次除法计算,则稀疏向量$\boldsymbol{\theta}_K$的计算复杂度由$O(K^2)$减小到$O(K)$.

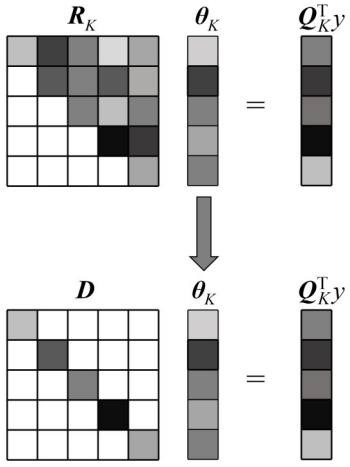


图3　稀疏向量$\boldsymbol{\theta}_K$的计算形式

在迭代计算中,为避免重复计算造成资源浪费,本文采用增量方式的WQR分解,即在每次迭中输入当前的匹配原子a_{λ_t},利用上次迭代$\boldsymbol{Q}_{t-1}$的结果计算$\boldsymbol{q}_t$,减少了$\boldsymbol{A}_t$的存储资源,并且避免了$\boldsymbol{Q}_{t-1}$重复计算[21,22]. 在使用矩阵$\boldsymbol{D}$计算稀疏向量$\boldsymbol{\theta}_K$时,由于各元素计算是相互独立的,因此每次迭代稀疏向量$\boldsymbol{\theta}_K$的每个元素可以表示为

$$\hat{\theta}_{\lambda_t}=\frac{\boldsymbol{q}_t^{\mathrm{T}}y}{d_t} \tag{15}$$

其中,$\hat{\theta}_{\lambda_t}$是$\hat{\boldsymbol{\theta}}(\hat{\boldsymbol{\theta}}\in R^N)$索引号为$\lambda_t$的元素;$d_t$为$\boldsymbol{D}$的第$t$个对角元素,即$\boldsymbol{R}_K$主对角线上第$t$个元素. 在每次迭代中,计算当前迭代稀疏向量的元素,减少矩阵$\boldsymbol{D}$存储消耗. 将WQR分解的优化方法与OMP算法结合,得到WQR-OMP算法计算流程如算法2所示.

3.2　复杂度与误差分析

计算复杂度和存储资源是评价算法性能的两个重要指标[23]. 在优化OMP算法时,QR分解将新的原子集$\boldsymbol{A}_t$分解成矩阵$\boldsymbol{Q}$和矩阵$\boldsymbol{R}$,然后带入最小二乘法计算中避免矩阵直接求逆的计算,降低计算复杂度,但这同时增加了矩阵$\boldsymbol{Q}$和矩阵$\boldsymbol{R}$的存储消耗. 而本文提出的WQR-OMP算法是在QR分解的基础上,对矩阵$\boldsymbol{R}$优化,避免$\boldsymbol{R}$矩阵的存储,同时降低了稀疏向量的计算复杂

算法2 WQR-OMP

输入: 测量值$y\in\mathbf{R}^{M\times 1}$;传感矩阵$\boldsymbol{A}=\boldsymbol{\Phi\Psi},\boldsymbol{A}\in\mathbf{R}^{M\times N}$;稀疏度$K$

输出: 重构信号$\hat{x}\in\mathbf{R}^{N\times 1}$

算法流程:

1. 初始化:$t=1,r_0=y,\boldsymbol{Q}_0=\varnothing,\hat{\boldsymbol{\theta}}=0$

2. 寻找最大匹配原子列的索引号:$\lambda_t=\arg\max\limits_{1\leqslant j\leqslant N}\left|\left\langle r_{t-1},\boldsymbol{a}_j\right\rangle\right|$

3. WQR分解:FOR $i=1,2,\cdots,t$

$$w=\sum_{i=2}^{t}\left(\boldsymbol{a}_{\lambda_t}^{\mathrm{T}}\boldsymbol{q}_{i-1}\right)\boldsymbol{q}_{i-1}$$

END FOR

$$q_t=a_{\lambda_t}-w$$

$$d_t=\left\|\boldsymbol{q}_t\right\|$$

$$\boldsymbol{q}_t=\frac{\boldsymbol{q}_t}{d_t}$$

4. 更新矩阵$\boldsymbol{Q}$:$\boldsymbol{Q}_t=\boldsymbol{Q}_{t-1}\cup\left\{\boldsymbol{q}_t\right\}$

5. 稀疏向量:$\hat{\boldsymbol{\theta}}_{\lambda_t}=\dfrac{\boldsymbol{q}_t^{\mathrm{T}}y}{d_t}$

6. 更新残差:$r_t=r_{t-1}-\boldsymbol{q}_t\boldsymbol{q}_t^{\mathrm{T}}y$

7. 迭代判断:$t=t+1$,若$t\leqslant K$,执行流程2,否则执行流程8

8. 利用稀疏基求得重构信号,即$\hat{x}=\boldsymbol{\Psi}\hat{\boldsymbol{\theta}}$

度. 另外,Batch-OMP算法是针对同一传感矩阵批量信号重建的OMP优化算法. 其原理是通过预先计算和Cholesky分解的方法降低计算复杂度. 为客观地评价WQR-OMP算法性能,将它分别与QR-OMP算法和Batch-OMP算法比较,结果如表1所示. 对比计算复杂度:优化后的OMP算法计算复杂度均为$O(K^3)$,Batch-OMP算法的计算复杂度优于QR-OMP算法和WQR-OMP算法,WQR-OMP算法的计算复杂度略小于QR-OMP算法. 对比存储资源:QR-OMP算法和Batch-OMP算法的存储资源为$O(K^2)$,而WQR-OMP算法的存储资源为$O(K)$,可见WQR-OMP算法对存储资源的要求较低. 例如,假设$K=M/4$,$M=N/4$,则QR-OMP算法、Batch-OMP算法和WQR-OMP算法的计算复杂度分别约为$66K^3$,$5K^3+128K^2$,$66K^3-65K^2$,存储资源分别约为$69K^2$,$266K^2$,$68K^2$. 通过比较分析,WQR-OMP算法的计算复杂度介于QR-OMP算法与Batch-OMP算法之间,但存储资源相比QR-OMP算法和Batch-OMP算法消耗较少. 因此,本文提出的WQR-OMP算法适用于资源受限的硬件平台实现.

表1 算法复杂度比较

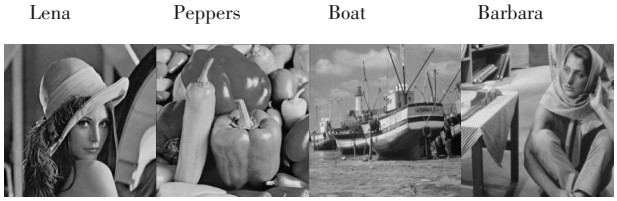
算法	计算复杂度	存储资源
OMP[23]	$\frac{K^4}{12}+\frac{2M}{3}K^3+2MK^2+NMK$	$2K+3MN+2M$
QR-OMP[23]	$\frac{2K^3}{6}+\frac{M}{2}K^2+NMK$	$K^2+MK+MN$
Batch-OMP[11,23]	K^3+K^2M+2MN	K^2+MK+N^2
WQR-OMP	$\frac{2K^3}{6}+\frac{M-2}{2}K^2+(NM-1)K$	$MK+MN$

由于权重化运算会产生一定的误差,为了分析误差对信号重构质量的影响,利用Matlab软件进行仿真. 在采样率M/N为0.25且稀疏度K为15的情况下,分别使用QR-OMP算法、Batch-OMP算法和WQR-OMP算法对图4(a)中256×256分辨率的"Lena""Pepper""Boat"和"Barbara"四幅图像进行重构,重构图像如图4(b)~(d)所示. 通过观察,三种OMP优化算法都能重构出图像,并且重构图像效果几乎一致. 为了分析重构图像之间的差异,使用峰值信噪比(Peak Signal to Noise Ration,PSNR)和结构相似性(Structural Similarity,SSIM)两个指标对图像重构质量进行比较,结果如表2所示. 对比同一幅图像,三种算法重构图像的PSNR相差小于0.7,SSIM相差小于0.07,可见三种算法图像重构质量差异并不显著. 在需要快速成像的领域,图像质量的轻微损失是可以接受的. 综上所述,本文提出WQR-OMP算法在降低计算复杂度和减少存储资源的同时,能够保证信号的重构质量.

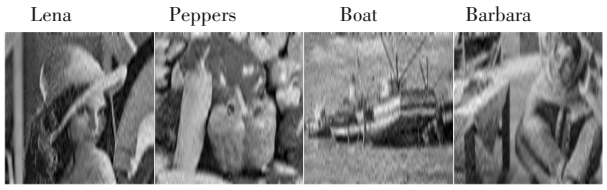
3.3 WQR-OMP SOC设计

在Xilinx Vivado HLS开发工具中,使用C++语言设计WQR-OMP算法硬件结构,然后综合把C++语言转化为RTL电路结构,最终封装成WQR-OMP IP核. 通过调用WQR-OMP IP,在ZYNQ 7020型号芯片上搭建WQR-OMP SOC系统. 图5为WQR-OMP SOC系统原理图. 该系统主要由硬件电路和软件程序组成. 硬件电路主要包括ZYNQ处理器IP、WQR-OMP IP、系统复位IP和AXI总线互联IP. 其中ZYNQ处理器主频为666 MHz ARM Cortex-A9,给外围电路提供100 MHz的时钟频率;软件程序主要包括硬件驱动程序和控制程序,在Xilinx SDK软件开发工具中实现,最终加载到ZYNQ处理器上运行. 工作原理:首先ARM读取SDK中的测试数据并写入DDR中;然后WQR-OMP IP通过M_AXI接口连接到AXI总线互联IP,从AXI_HP接口读取DDR上的测试数据进行运算;最后运算结束后输出结果写入DDR,并通过UART打印到PC端.

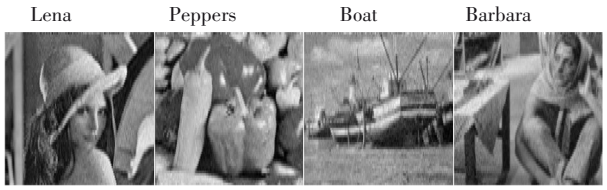
为了提高数值的动态范围和确保数值计算的精



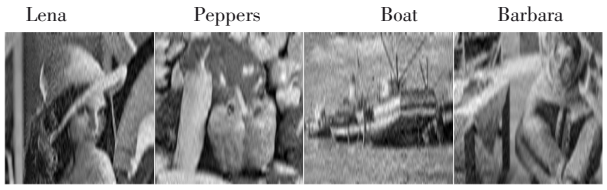
(*a*) 原图



(*b*) QR-OMP算法重构图像



(*c*) Batch-OMP算法重构图像



(*d*) WQR-OMP算法重构图像

图4　图像重构结果

表2　三种算法重建质量比较

图像	评价指标	QR-OMP	Batch-OMP	WQR-OMP
Lena	PSNR/dB	22.590	22.131	22.572
	SSIM	0.426	0.481	0.425
Peppers	PSNR/dB	23.298	23.207	23.285
	SSIM	0.426	0.449	0.423
Boat	PSNR/dB	22.360	22.936	22.372
	SSIM	0.330	0.391	0.329
Barbara	PSNR/dB	23.415	23.998	23.386
	SSIM	0.452	0.491	0.450

度,本文在设计WQR-OMP算法硬件结构时,数据类型采用单精度浮点型. 图6为WQR-OMP算法硬件结构图. 该结构主要包括存储单元和计算单元. 其中计算单元由最大匹配原子、WQR分解、稀疏向量和更新残差计算单元组成. 该结构的计算流程:第一步,传感矩阵 $\boldsymbol{A}$ 与残差 r_{t-1} 先经过最大匹配原子计算单元运算,然后输出列序号 λ_t;第二步,λ_t 把对应列 a_{λ_t} 输入WQR分解计算单元,经运算输出 $\boldsymbol{q}_t$ 和 d_t,然后更新 $\boldsymbol{Q}$;第三步,输入 $\boldsymbol{q}_t$,d_t 和 y 到稀疏向量计算单元,经运算输出 $\hat{\boldsymbol{\theta}}_{\lambda_t}$ 和 $\boldsymbol{q}_t^{\mathrm{T}}y$,然后更新 $\hat{\boldsymbol{\theta}}$;第四步,输入 $\boldsymbol{q}_t$、$\boldsymbol{q}_t^{\mathrm{T}}y$ 和 r_{t-1} 到更新残差计算单元,经运算输出 r_t,然后更新 r 并执行下一次迭代. 其中最大匹配原子和WQR分解两个计算单元涉及向量内积计算,所以优化向量内积计算是提高计算效率的关键之一.

传统的向量内积计算一般形式为 $\mathrm{sum}=\mathrm{sum}+a[i]\times b[i]$,主要通过循环乘累加方式计算结果,时间延迟较大. 为了提高向量内积计算效率,使用流水化指令PIPELINE和数组分区指令ARRAY_PARTITION,降低计算延迟,增大数据吞吐率,实现向量内积并行乘法计算和加法树的硬件结构. 利用Xilinx Vivado HLS开发工具综合分析,随着数组分区指令ARRAY_PARTION的分区因子factor增加,资源消耗不断增加,时间延迟不断减少. 考虑资源消耗与时间延迟两方面的因素,最终分区因子factor设为8,即一个时钟周期读取8个数组元素实现并行计算的硬件结构.

4　实验与结果

4.1　算法硬件结构仿真

在信号长度 N 为256、采样率为0.25、稀疏度 K 为15的条件下,使用Xilinx Vivado HLS开发工具,分别设计QR-OMP、Batch-OMP和WQR-OMP三种算法的硬件结构,并生成综合报告,比较三种算法硬件结构的重构时间和资源消耗. 表3与表4分别列出三种算法硬件结构的重构时间和资源消耗.

在表3中重构时间主要由数据传输和数据计算两部分组成. 对比数据传输时间,QR-OMP算法和WQR-OMP算法需要输入与输出的数据为 y、$\boldsymbol{A}$ 和 $\hat{\boldsymbol{\theta}}$,数据传输个数为 $M+MN+N$,数据传输时钟周期约为16 714. Batch-OMP算法需要输入与输出的数据为 $h=\boldsymbol{A}^{\mathrm{T}}y$、$\boldsymbol{G}=\boldsymbol{A}^{\mathrm{T}}\boldsymbol{A}$ 和 $\hat{\boldsymbol{\theta}}$,数据传输个数为 $2N+N^2$,数据传输时钟周期约为66 058. 对比数据计算时间,QR-OMP、Batch-OMP和WQR-OMP三种算法硬件结构计算所需的时钟周期分别为56 944、19 184和35 565,则说明WQR-OMP算法的计算复杂度低于QR-OMP算法但高于Batch-OMP算法,与表1中计算复杂度的分析一致. 通过分析比较,WQR-OMP算法硬件结构总的重构时间显著小于QR-OMP算法和Batch-OMP算法硬件结构,可见WQR-OMP算法硬件结构的运行效率更高.

表4是三种算法硬件结构资源消耗的比较. Batch-OMP算法硬件结构的资源消耗明显多于QR-OMP算法和WQR-OMP算法硬件结构. 另外,WQR-OMP算法硬件结构的BRAM_18K、FF和LUT资源消耗少于QR-OMP算法硬件结构. 结果表明,WQR-OMP算法硬件结

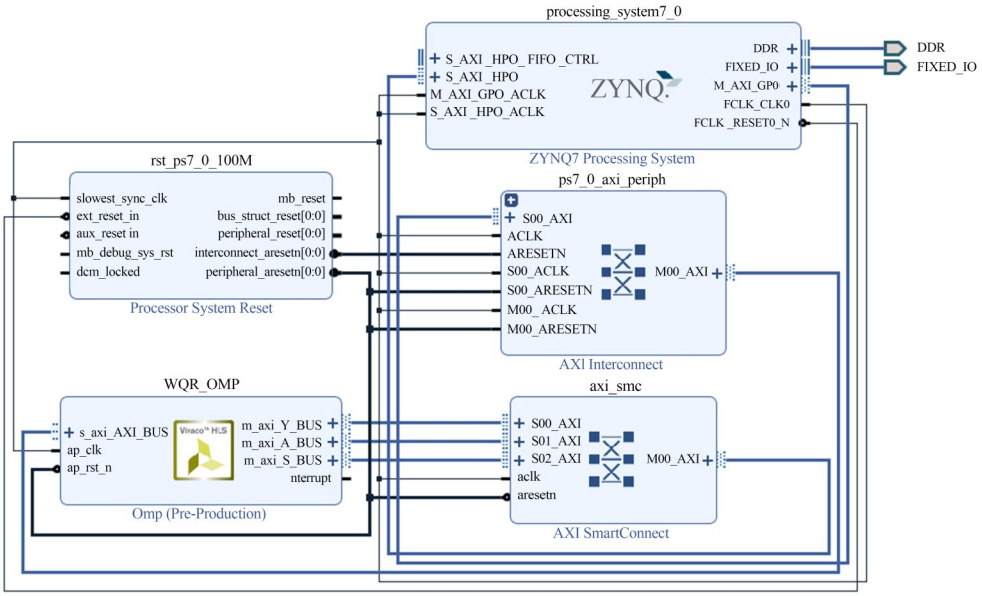


图5 WQR-OMP SOC原理图

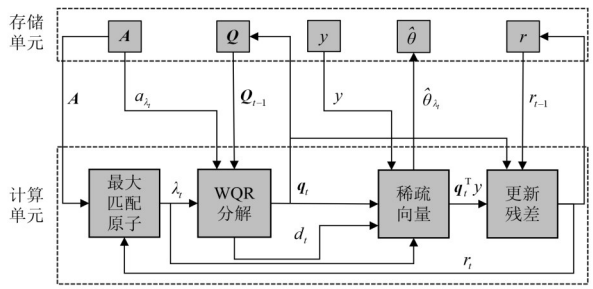


图6 WQR-OMP算法硬件结构

表3 重建时间估计

时钟周期(Latency)	QR-OMP	Batch-OMP	WQR-OMP
数据传输	16 714	66 058	16 714
数据计算	56 944	19 184	35 565
合计	73 659	85 242	52 279

表4 资源消耗

资源	QR-OMP	Batch-OMP	WQR-OMP
BRAM_18K	88	170	87
DSP48E	40	97	40
FF	19 324	18 836	18 928
LUT	13 165	23 374	13 021

构资源消耗较少,节省了硬件成本的开销.

综上所述,与QR-OMP算法和Batch-OMP算法的硬件结构相比,WQR-OMP算法硬件结构具有重构速度快、资源消耗少的优势,更适用于小型化、低成本的硬件平台实现.

4.2 WQR-OMP SOC重构结果

本文设计的WQR-OMP SOC系统资源消耗情况如表5所示,其中DSP消耗40个,BRAM消耗42个,资源消耗较少. WQR-OMP SOC系统动态功率和静态功率分别2.109 W和0.160 W. 动态功率主要由ARM处理器和WQR-OMP IP组成,其中ARM处理器占了73%.

表5 WQR-OMP SOC资源消耗

资源	消耗/个	总资源/个	利用率/%
LUT	16 773	53 200	31.53
LUTRAM	1 578	17 400	9.07
FF	19 446	106 400	18.28
BRAM	42	140	30.00
DSP	40	220	18.18
BUFG	1	32	3.13

为验证WQR-OMP SOC系统的加速效果,利用ARM处理器设计WQR-OMP算法,然后分别使用WQR-OMP SOC和ARM处理器对图4(*a*)中的四幅图像进行重构,重构结果如表6所示. 对比重构速度,WQR-OMP SOC的重构时间约为400 ms,而ARM处理器重构时间约为2 911 ms,则WQR-OMP SOC的重构速率相比ARM处理器的重构速率提升了约6.3倍.

4.3 与其他硬件实现工作对比

在测量矩阵维度和稀疏度相同的条件下,将本文的设计与其他研究者的工作相比,如表7所示. 在与文献[10]对比时,本文工作频率为100 MHz,Block RAM消耗38个,DSP消耗35个,LUT消耗15.0K个,重构时间为1.420 ms,而文献[10]工作频率为113 MHz,Block RAM消耗132个,DSP消耗446个,LUT消耗114.0K个,重构时间为0.021 ms. 虽然本文重构速度较慢,但资源

表6　图像重构时间对比

图像	WQR-OMP SOC/ms	ARM/ms
Lena (PSNR=22.586 dB, SSIM=0.426)	399.755	2 911.565
Peppers (PSNR=23.338 dB, SSIM=0.423)	399.788	2 911.673
Boat (PSNR=22.388 dB, SSIM=0.327)	400.249	2 911.853
Barbara (PSNR=23.384 dB, SSIM=0.450)	399.787	2 911.621

消耗远少于文献[10],硬件实现的成本更低. 与文献[24]对比时,本文工作频率为100 MHz,Block RAM消耗44个,DSP消耗57个,LUT消耗18.5K个,重构时间为2.075 ms. 文献[24]工作频率为110 MHz,Block RAM消耗81个,DSP消耗36个,LUT消耗13.7K个,重构时间为2.500 ms. 虽然本文的DSP和LUT资源消耗多于文献[24],但工作频率和Block RAM资源消耗都较低,并且重构速度也较快. 通过分析,重构速度受到硬件平台资源和工作频率的影响. 在对重构速度要求高的应用场景中,可以以资源和工作频率为代价,提高重构速度. 而本文设计的WQR-OMP SOC系统具有存储资源消耗较少、重构速度较快的优势,适用于低成本的硬件平台.

表7　与其他研究者工作比较

测量矩阵维度$M\times N$	稀疏度K	对比文献	硬件平台	工作频率/MHz	重构时间/ms	Block RAM/个	DSP/个	LUT/个
64×256	8	文献[10]	Zynq UltraScale	113	0.021	132	446	114.0K
		本文	Zynq 7020	100	1.420	38	35	15.0K
84×256	15	文献[24]	Zed Board	110	2.500	81	36	13.7K
		本文	Zynq 7020	100	2.075	44	57	18.5K

5　结论

本文使用基于Haar小波稀疏排序的Hadamard矩阵与DCT稀疏基构造一种具有特殊分布的传感矩阵$\boldsymbol{A}$,对OMP算法进一步优化,提出一种低复杂度、低成本的WQR-OMP算法硬件结构. 其原理是根据传感矩阵$\boldsymbol{A}$的QR分解中矩阵$\boldsymbol{R}$元素的分布特性,通过单位权重化矩阵$\boldsymbol{W}$与矩阵$\boldsymbol{R}$进行运算,只保留主对角线上的元素,其余元素清零,得到一个对角矩阵$\boldsymbol{D}$,然后近似计算稀疏向量,从而降低OMP算法的计算复杂度并减少存储资源的消耗,有效地解决了QR分解增加存储资源的问题. 在Xilinx Vivado HLS开发工具中,设计了WQR-OMP算法硬件结构,最终在ZYNQ 7020型号芯片上搭建了WQR-OMP SOC系统. 实验结果表明,本文提出的WQR-OMP算法硬件结构的重构速度和资源消耗均优于QR-OMP算法和Batch-OMP算法硬件结构. 在压缩率为0.25的条件下,WQR-OMP SOC系统对256×256分辨率图像的重构时间为400 ms左右,其重构速率比仅使用ARM处理器的重构速率提高了约6.3倍. 与其他现有研究相比,该系统使用Block RAM存储资源较少,且重构速率有一定的提升,适用于存储资源受限的硬件平台.

参考文献

[1] DONOHO D L. Compressed sensing[J]. IEEE Transactions on Information Theory, 2006, 52(4): 1289-1306.

[2] CANDES E J, TAO T. Near-optimal signal recovery from random projections: Universal encoding strategies?[J]. IEEE Transactions on Information Theory, 2006, 52(12): 5406-5425.

[3] 左婷, 王法松, 张建康, 等. 室内可见光通信系统中基于压缩感知的空移键控信号检测方法[J]. 电子学报, 2022, 50(1): 36-44.
ZUO T, WANG F S, ZHANG J K, et al. Space shift keying signal detection approach based on compressed sensing in indoor VLC system[J]. Acta Electronica Sinica, 2022, 50 (1): 36-44. (in Chinese)

[4] LUSTIG M, DONOHO D L, SANTOS J M, et al. Compressed sensing MRI[J]. IEEE Signal Processing Magazine, 2008, 25(2): 72-82.

[5] 肖许意, 陈刘雅, 张学智, 等. 单像素成像及其概率统计分析综述[J]. 激光与光电子学进展, 2021, 58(10): 1011018.
XIAO X Y, CHEN L Y, ZHANG X Z, et al. Review on single-pixel imaging and its probability statistical analysis [J]. Laser & Optoelectronics Progress, 2021, 58(10): 1011018. (in Chinese)

[6] CRESPO MARQUES E, MACIEL N, NAVINER L, et al. A review of sparse recovery algorithms[J]. IEEE Access, 2018, 7: 1300-1322.

[7] POLAT Ö, KAYHAN S K. High-speed FPGA implementation of orthogonal matching pursuit for compressive sensing signal reconstruction[J]. Computers & Electrical Engineering, 2018, 71: 173-190.

[8] RABAH H, AMIRA A, MOHANTY B K, et al. FPGA implementation of orthogonal matching pursuit for compres-

sive sensing reconstruction[J]. IEEE Transactions on Very Large Scale Integration (VLSI) Systems, 2015, 23(10): 2209-2220.
[9] GE X, YANG F, ZHU H L, et al. An efficient FPGA implementation of orthogonal matching pursuit with square-root-free QR decomposition[J]. IEEE Transactions on Very Large Scale Integration (VLSI) Systems, 2019, 27(3): 611-623.
[10] LI J, CHOW P, PENG Y X, et al. FPGA implementation of an improved OMP for compressive sensing reconstruction[J]. IEEE Transactions on Very Large Scale Integration (VLSI) Systems, 2021, 29(2): 259-272.
[11] RUBINSTEIN R, ZIBULEVSKY M, ELAD M. Efficient implementation of the K-SVD algorithm using batch orthogonal matching pursuit[EB/OL]. (2008)[2022]. http://cs.technion.ac.il/users/wwwb/cgi-bin/tr-get.cgi/2008/CS/CS-2008-08.revised.pdf.
[12] ZHANG F, PIAO Y. Design of restoration method based on compressed sensing and TwIST algorithm[J]. Journal of Physics: Conference Series, 2018, 1004: 012006.
[13] BARANIUK R G. Compressive sensing[lecture notes][J]. IEEE Signal Processing Magazine, 2007, 24(4): 118-121.
[14] LI L X, FANG Y, LIU L W, et al. Overview of compressed sensing: Sensing model, reconstruction algorithm, and its applications[J]. Applied Sciences, 2020, 10(17): 5909.
[15] CANDES E J, ROMBERG J, TAO T. Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information[J]. IEEE Transactions on Information Theory, 2006, 52(2): 489-509.
[16] DONOHO D L, ELAD M. Optimally sparse representation in general (nonorthogonal) dictionaries via l minimization[J]. Proceedings of the National Academy of Sciences of the United States of America, 2003, 100(5): 2197-2202.
[17] TROPP J A, GILBERT A C. Signal recovery from random measurements via orthogonal matching pursuit[J]. IEEE Transactions on Information Theory, 2007, 53(12): 4655-4666.
[18] 李佳, 王强, 沈毅, 等. 压缩感知中测量矩阵与重建算法的协同构造[J]. 电子学报, 2013, 41(1): 29-34.
LI J, WANG Q, SHEN Y, et al. Collaborative construction of measurement matrix and reconstruction algorithm in compressive sensing[J]. Acta Electronica Sinica, 2013, 41(1): 29-34. (in Chinese)
[19] YU Z L, SU J C, YANG F, et al. Fast compressive sensing reconstruction algorithm on FPGA using Orthogonal Matching Pursuit[C]//2016 IEEE International Symposium on Circuits and Systems (ISCAS). Piscataway: IEEE, 2016: 249-252.
[20] 李明飞, 阎璐, 杨然, 等. 基于Hadamard矩阵优化排序的快速单像素成像[J]. 物理学报, 2019, 68(6): 87-94.
LI M F, YAN L, YANG R, et al. Fast single-pixel imaging based on optimized reordering Hadamard basis[J]. Acta Physica Sinica, 2019, 68(6): 87-94. (in Chinese)
[21] GHARAVI-ALKHANSARI M, HUANG T S. A fast orthogonal matching pursuit algorithm[C]//Proceedings of the 1998 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP'98 (Cat. No.98CH36181). Piscataway: IEEE, 2002: 1389-1392.
[22] ROY S, ACHARYA D P, SAHOO A K. Low-complexity architecture of orthogonal matching pursuit based on QR decomposition[J]. IEEE Transactions on Very Large Scale Integration (VLSI) Systems, 2019, 27(7): 1623-1632.
[23] FARDAD M, SAYEDI S M, YAZDIAN E. A low-complexity hardware for deterministic compressive sensing reconstruction[J]. IEEE Transactions on Circuits and Systems I: Regular Papers, 2018, 65(10): 3349-3361.
[24] KOPPARTHI V R, PEESAPATI R, SABAT S L. System on chip implementation of low complex orthogonal matching pursuit algorithm on FPGA[C]//2020 6th International Conference on Signal Processing and Communication (ICSC). Piscataway: IEEE, 2020: 178-184.

作者简介

王 玺 男,1983年出生,重庆人. 2013年6月于重庆大学获得工学博士学位. 2013年3月至2018年8月就职于中国电子科技集团第四十四研究所,任高级工程师. 2018年9月至今就职于重庆邮电大学,任硕士生导师. 主要研究方向为稀疏优化方法及其硬件加速计算.
E-mail: xiwang@cqupt.edu.cn

梁文凯 男,1997年出生,河南周口人. 重庆邮电大学电子科学与技术专业硕士研究生. 主要研究方向为压缩感知算法的研究及硬件实现.
E-mail: 1429205353@qq.com

黎 森 男,1982年出生,重庆人. 工学博士. 副教授,硕士生导师. 主要研究方向为X射线及Gamma射线超快诊断技术.
E-mail: limiao@cqupt.edu.cn