

基于粒子群优化的稀疏分解最优匹配原子搜索算法*

王春光¹, 刘金江², 孙即祥¹

(1. 国防科技大学 电子科学与工程学院, 湖南 长沙 410073; 2. 南阳师范学院 计算机系, 河南 南阳 473061)

摘要: 信号的稀疏分解能得到信号的稀疏表示形式, 便于进一步处理, 但其计算非常复杂, 是一个 NP 问题。粒子群优化是群体智能优化算法, 算法简单, 易于实现, 且搜索效果好。把粒子群优化算法用于稀疏分解的最优匹配原子的搜索, 能降低稀疏分解复杂度, 同时减少稀疏分解的超完备字典对存储空间的占用, 以提高用稀疏分解理论进行信号处理的计算效率, 满足或接近实时性的要求。实验证明, 此方法切实可行。

关键词: 粒子群优化; 稀疏分解; 心电信号; 图像处理

中图分类号: TN911.72 文献标识码: A

Algorithm of Searching for the Best Matching Atoms Based on Particle Swarm Optimization in Sparse Decomposition

WANG Chun-guang¹, LIU Jin-jiang², SUN Ji-xiang¹

(1. College of Electronic and Engineering, National Univ. of Defense Technology, Changsha 410073, China;

2. Department of Computer, Nanyang Normal University, Nanyang 473061, China)

Abstract: Sparse decomposition of signal can get sparse representation of signal, and then next disposal can use this sparse representation expediently. But sparse decomposition is very complex (NP problem). Particle swarm optimization is a kind of optimization algorithm using colony aptitude. Its theory is simple to be realized, and the result of searching is good. To reduce complexity of sparse decomposition and space of memory, particle swarm optimization is used in searching the best atom. Particle swarm optimization can increase the efficiency processing signal using sparse decomposition, and then this method can meet (or near) the request of real time. Examinations have proved that this method is feasible.

Key words: particle swarm optimization; sparse decomposition; ECG; image processin

作为一种新的信号处理理论, 稀疏分解一出现就引起极大的关注^[1]。其在信号处理的各个领域的研究在短时间内都得到了长足的发展。但是稀疏分解的基是超完备的, 基中原子数远远大于信号长度, 而稀疏分解的目的就是从超完备基中搜索出最符合信号特征的那些原子来近似(或精确)表示信号, 这个搜索过程无论是基于全局搜索的贪婪算法^[2](如 MP、OMP), 还是基于凸松弛和线性规划的基寻踪算法(BP)^[3], 都是 NP 问题。因此稀疏分解算法的复杂度非常高, 对计算速度和内存的要求非常高, 这也限制了稀疏分解的实际应用。采用优化稀疏分解中最佳原子的搜索算法是降低计算复杂度的途径。

粒子群优化^[4]是模拟生物群体模型(鸟群)的群体智能优化算法, 算法赋予每个粒子一定的速度、个体能力和社会能力, 通过追踪个体最优值和群体最优值, 实现全局搜索。算法简单、容易实现, 需调整参数少, 得到广泛应用^[5]。

将粒子群优化算法用于稀疏分解最优原子搜索, 可以加快搜索速度, 减少内存的占用, 是降低稀疏分解复杂度的一个新思路。

1 稀疏分解和粒子群优化的基本原理

1.1 稀疏分解

用稀疏的数据表示原始数据对信号处理非常有利。传统的信号表示理论多基于完备正交基, 但实

* 收稿日期: 2007-09-04

作者简介: 王春光(1975—), 男, 助理工程师, 博士生。

际上许多信号是各种自然现象的混合,在正交变换中无法找到这些混合信号的稀疏表示方法。稀疏分解是基(字典)超完备的,可以根据信号的特点选择最逼近信号结构的原子作为基(字典),从字典中搜索出与信号最近似的原子,用这些原子及其相应的分解系数近似(或精确)地表示待分解信号。稀疏分解的数学描述为:

$$\min \|c\|_0 \quad \text{s.t.} \quad f = \sum_{k=0}^{M-1} c_k g_{v_k} \quad (1)$$

其中, f 为待分解信号, g_{v_k} 为超完备字典 \mathcal{D} 中的一个原子, c 为解向量。与正交分解相比,稀疏分解更接近人类的主观判断方法。

1.2 粒子群优化

m 个粒子(Particle)组成群体(Swarm),每个粒子在 D 维搜索空间中以一定的速度飞行,每个粒子根据自己搜索到的历史最好点和群体内粒子的历史最好点,更新位置。

第 i 个粒子的位置表示为: $x_i = \begin{Bmatrix} x_{i1}, x_{i2}, \dots, x_{id}, \dots, x_{iD} \end{Bmatrix}$, $i = 1, 2, \dots, m$ 。

第 i 个粒子的速度表示为: $v_i = \begin{Bmatrix} v_{i1}, v_{i2}, \dots, v_{id}, \dots, v_{iD} \end{Bmatrix}$, $i = 1, 2, \dots, m$ 。

第 i 个粒子的经历过的历史最好点表示为: $P_i = \begin{Bmatrix} p_{i1}, p_{i2}, \dots, p_{id}, \dots, p_{iD} \end{Bmatrix}$, $i = 1, 2, \dots, m$ 。

群体内所有粒子所经过的最好的点表示为: $P_g = \begin{Bmatrix} p_{g1}, p_{g2}, \dots, p_{gd}, \dots, p_{gD} \end{Bmatrix}$ 。

粒子的位置和速度根据如下方程进行变化:

$$v_{id}^{k+1} = \omega v_{id}^k + c_1 \xi \left(p_{id}^k - x_{id}^k \right) + c_2 \eta \left(p_{gd}^k - x_{id}^k \right) \quad (2)$$

$$x_{id}^{k+1} = x_{id}^k + v_{id}^{k+1} \quad (3)$$

这里, ω 为惯性因子,决定着粒子对当前速度继承的大小。 c_1 和 c_2 为正常数,统称学习因子。学习因子使粒子具有自我总结和向群体中优秀个体学习的能力,从而向自己的历史最优点以及群体内的历史最优点靠近。 ξ 和 η 是在 $[0, 1]$ 区间内均匀分布的伪随机数。粒子的速度被限制在一个最大速度 v_{\max} = $\begin{Bmatrix} v_{\max}^1, v_{\max}^2, \dots, v_{\max}^D \end{Bmatrix}$ 的范围内。

2 用粒子群优化进行最优原子搜索

本文算法可以看作从贪婪算法发展而来。在最优原子搜索中,常用的贪婪算法包括匹配追踪(MP)、正交匹配算法(OMP)及其变种。字典的超完备性和全局搜索是造成使用贪婪算法搜索匹配原子的稀疏分解计算复杂且占用大量内存的主因。以常用的 Gabor 字典^[6]为例说明这一情况。Gabor 字典的生成函数为:

$$g_v(t) = \frac{1}{\sqrt{s}} g \left(\frac{t-u}{s} \right) \cos(vt + \omega) \quad (4)$$

其中, $g(t) = e^{-\pi t^2}$ 是高斯窗函数, $\gamma = (s, u, v, \omega)$ 是时频参数。时频参数按以下方法离散化: $\gamma = (a^j, pa^j \Delta u, ka^{-j} \Delta v, i \Delta \omega)$, 其中, $a = 2$, $\Delta u = 1/2$, $\Delta v = \pi$, $\Delta \omega = \pi/6$, $0 < j \leq \log_2 N$, $0 \leq p < 2^{-j+1} N$, $0 \leq k < 2^{j+1}$, $0 \leq i \leq 12$, N 为信号采样点数。因为生成函数为 Gabor 函数,此字典被称为 Gabor 字典。Gabor 字典冗余度很高,原子个数为 $52(N \log_2 N + N - 1)$ 。贪婪算法要遍历内积字典,并选取内积最大的原子,因此搜索算法的计算量将非常大。

用粒子群算法代替上述贪婪算法,能使搜索算法很快地收敛到全局的最优值(或次优值)。整个稀疏分解过程是一个迭代过程,迭代初期的某步搜索中所得原子即使不是全局最优值,而是一个局部最优值,也不会对整个分解造成不良影响,因为当前步中的未被选出最优原子很快会在随后的迭代过程中被选中。而迭代后期所选原子对信号的影响比较小,应该在后期被选出的原子即使迭代停止之前未被选出,也不会对信号分解造成太大的影响。

把超完备字典生成函数的参数作为粒子群优化算法参数搜索空间中的一维,搜索在参数空间中进行,这样就不用产生整个超完备字典,而是只产生种群中各粒子的参数向量来代表原

子,和整个超完备字典相比,种群中粒子参数向量占用的内存基本可以忽略不计。仍以 Gabor 字典来说,用粒子群优化实现最优原子搜索的这一特点。参照式(4),把生成函数的参数 s, u, v, ω 分别作为粒子群优化搜索空间中的一维,整个搜索空间为 (s, u, v, ω) 代表的四维空间。当待分解信号长度为 1024 时,设定 s, u 的范围为 $[0, 1024]$, v, ω 的范围为 $[0, 2\pi]$ 。标准粒子群优化算法是针对连续空间的,不需要对搜索空间离散化。上述 4 维参数空间能够生产的原子数是无穷的, Gabor 字典只是按其抽样规则抽样的结果。因此,用粒子群优化算法搜索空间的“词汇量”比 Gabor 字典更大,搜索得到的原子更能反映原信号的结构特点。

3 实验结果

3.1 一维心电信号(ECG)的实验结果

从 MIT-BIH 心电数据库中任取长度为 1024 个采样点(采样频率为 360Hz)的心电信号作为待分解信号(如图 1),采用第二部分中参数空间 (s, u, v, ω) 代表的超完备字典。粒子群优化的种群规模设定为 50 个粒子,迭代次数为 500,最大速度 $v_{\max} = (v_s, v_u, v_v, v_\omega) = (50, 50, 10, 10)$, ω 取 $[0.95, 0.4]$, ω 随迭代次数线性递减, c_1 和 c_2 取值为 2。

由先搜索到的 15 个原子重构的近似信号(如图 2)能量占原信号总能量的 95.96%,搜索到 40 个原子时,重构近似信号(如图 3)能量占原信号总能量的 97.59%,15 个原子以后的 25 个原子对原信号能量的总贡献只占到原信号总能量的 1.63%。从图 2、图 3 的对比,以及图 4 能量的分析图可知,总体而言,初期搜索到的原子对原信号能量的贡献较大且按搜索到的顺序迅速下降,后期搜索到的原子对原信号能量的贡献较低且趋于平缓。这证明了文中第 2 部分分析的粒子群优化当前步只能得到次优解对整个稀疏分解影响不大的论断的正确性。

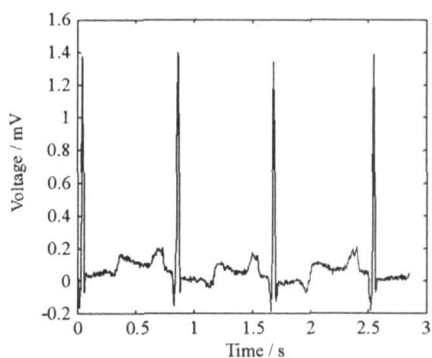


图 1 原心电图
Fig.1 Original ECG

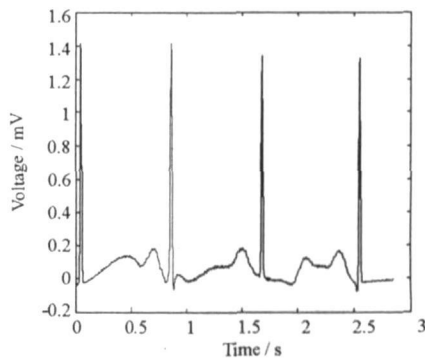


图 2 前 15 个原子重构图
Fig.2 Reconstruct ECG of 15 atoms

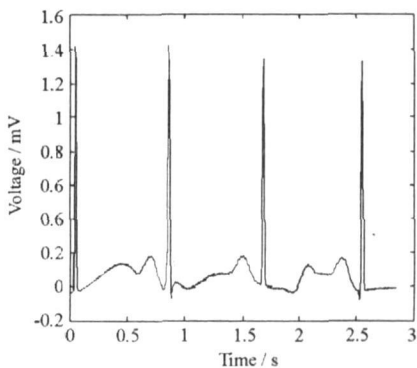


图 3 前 40 个原子重构图
Fig.3 Reconstruct ECG of 40 atoms

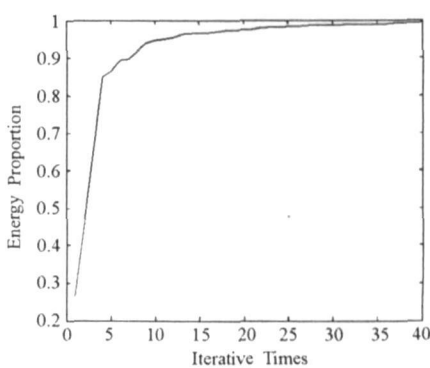


图 4 能量分析图
Fig.4 Energy analysis

在 Gabor 字典上,用匹配追踪算法(MP)搜索最优原子,每步迭代要遍历字典中所有原子(信号长度为 1024,则原子数为 585 676)。在相同条件下,用粒子群优化算法搜索最优原子,最多只需搜索 25 000 个参数空间点,再由这些点产生原子。相对于内积运算,粒子群优化中的速度和位置的更新以及根据点产生原子的运算都可以忽略不计。因此,粒子群优化算法复杂度就比匹配追踪算法至少下降 23 倍。粒子群优化算法不需要开辟额外的缓存空间用于存储字典,匹配追踪算法则需要开辟额外的缓存空间用于存储字典(包括 $52N(N \log_2 N + N - 1)$ 个浮点数),而且频繁地从缓存中读取数据也极大地影响运算速度。

考虑到高斯函数 99.75% 的能量集中于 $[-3\sigma, 3\sigma]$ (σ 为标准差),将内积运算的有效区域确定为原子的 $[-3\sqrt{2}\sigma/2+u, 3\sqrt{2}\sigma/2+u]$ 区域与剩余信号相应区域之间的内积。

粒子群优化过程中搜索到的原子是随机的,每次搜索用时也不尽相同,统计结果显示,用粒子群优化算法的稀疏分解算法的计算速度是用基本匹配搜索的稀疏分解算法速度的 135 倍左右。

3.2 二维图像的试验结果

根据图像特点选择合适的生成函数,对生成函数进行平移、旋转、伸缩、弯曲,产生超完备字典的原子,这些原子能体现图像的几何特征,在此字典的基础上可以得到图像的稀疏表示。目前,稀疏分解在图像处理中的应用研究较为广泛。

以用于图像压缩的超完备字典^[7]为例,说明本文提出的用粒子群优化算法搜索最优原子对图像稀疏分解的改进。为提高计算效率,文献[7]提出先用拉普拉斯金字塔算法把图像分成高、低频两个部分。低频部分用 DPCM 编码压缩,高频部分用稀疏分解得到图像边缘信息的稀疏表示,进而实现压缩。对低频我们不予讨论,文献[7]中有详细介绍。

高频部分稀疏分解的超完备字典的原子由两个生成函数产生,分别代表斜坡形和单峰形两种边缘。

$$g_1(x_1, x_2) = x_1 \exp\left[-\left(\frac{|x_1|}{\sigma} + x_2^4\right)\right] \quad (5)$$

$$g_2(x_1, x_2) = \left(4x_1^2 - 2\right) \exp\left[-\left(x_1^2 + x_2^2\right)\right] \quad (6)$$

平移算子: $\mathcal{T}_b g(\mathbf{x}) = g(\mathbf{x} - \mathbf{b})$, 其中 \mathbf{b} 为平移向量。

旋转算子: $\mathcal{R}_\theta g(\mathbf{x}) = g(\mathbf{y}_0(\mathbf{x}))$, 其中 $\mathbf{y}_0(\mathbf{x}) = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$, 为旋转矩阵, θ 为旋转角度。

弯曲算子: $\mathcal{B}_r g(x_1, x_2) = \begin{cases} g\left(r - \sqrt{(x_1 - r)^2 + x_2^2}, \arctan\left(\frac{x_2}{r - x_1}\right)\right), & x_1 < r \\ g\left(r - |x_2|, x_1 - r + r \frac{\pi}{2}\right), & x_1 \geq r \end{cases}$, 其中 r 为弯曲半径。

伸缩算子: $\mathcal{S}_a g(\mathbf{x}) = \mathcal{S}_{a_1, a_2} g(x_1, x_2) = g\left(\frac{x_1}{a_1}, \frac{x_2}{a_2}\right)$, 其中 \mathbf{a} 为尺度向量。

以上四种算子作用于两个生成函数,产生未归一化的原子,经归一化就得到超完备字典的原子。

$$\text{atom}_{(b, \theta, r, a)}(\mathbf{x}) = \frac{\mathcal{T}_b \mathcal{R}_\theta \mathcal{B}_r \mathcal{S}_a g(\mathbf{x})}{\|\mathcal{T}_b \mathcal{R}_\theta \mathcal{B}_r \mathcal{S}_a g(\mathbf{x})\|} \quad (7)$$

整个超完备字典由 g_1, g_2 两个生成函数生成的子超完备字典构成,分别代表斜坡形和单峰形两种边缘。粒子群优化算法的搜索空间就是四种算子参数构成的参数空间 $(b_1, b_2, \theta, r, a_1, a_2)$ 。

以 256×256 像素 Lena 标准图为例。对该图做两层的拉普拉斯金字塔分解,得到两个高频图像。用第二层高频图像 (128×128) 作为实验数据(如图 5 左图),说明图像的稀疏分解中用粒子群优化算法搜索最优匹配原子的算法的可行性。参数搜索空间上限设定为 $(128, 128, 360, 128, 16, 16)$, 下限设定为 $(0, 0, 0, 0, 0, 0)$ 。种群规模为 20, 优化迭代次数为 500 次,最大速度为 $(20, 20, 30, 20, 4, 4)$ 。 ω 取 $[0.95, 0.4]$, 随迭代次数线性递减, c_1 和 c_2 取值为 2。

用搜索到的 300 个匹配原子重构高频图,如图 5 右图。

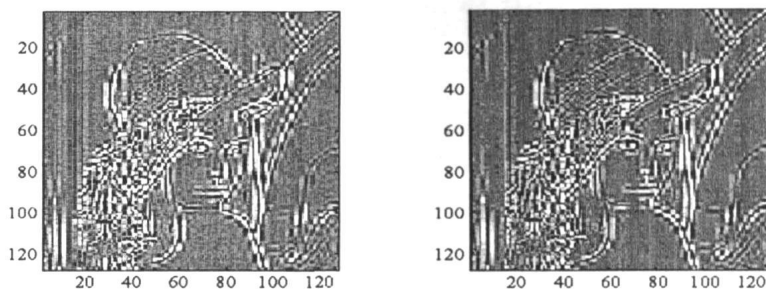


图5 左图为Lena第二层高频成分,右图为300个原子的重构高频图

Fig. 5 Left figure is 2nd HF composition, right figure is 2nd HF composition reconstructed by 300 atoms

与一维信号相似,粒子群优化算法的参数搜索空间不需要离散化,整个参数空间代表的字典中的原子个数是无限大的,比匹配追踪算法中使用的离散化的字典的“词汇量”更大,更能准确表示图像特征。同样,搜索在参数空间中进行,字典不会占用额外的内存。当前步搜索只能找到次优解也不会对最终结果产生太大的影响。原子对高频成分的能量贡献与一维信号相似,限于篇幅不再重复。

4 算法讨论和总结

信号的稀疏分解作为一种新发展起来的信号变换方法,以其表示信号特征的准确性和稀疏性引起人们的浓厚兴趣,迅速发展成为信号处理中的热点,研究人员对其发展前景非常看好。对稀疏分解理论研究和实践应用都做出重大贡献的M. Elad说:“The dream: the over-completeness idea is highly effective, and should replace existing methods in representation and inverse-problems.”^[8]

然而,计算复杂度始终是信号的稀疏分解实际应用的一个瓶颈,特别是对实时性要求较高时。本文把粒子群优化算法应用于稀疏分解最优原子的搜索,降低了计算复杂度,节省了内存,部分解决了算法实时性差的问题。这种最优匹配原子的搜索算法可以很方便和其他快速算法结合使用(如分块降维处理的方法或插值降维处理的方法等),能够进一步降低稀疏分解的计算复杂度。原因是这些快速算法对实验数据降维后使用的搜索算法本质上还是贪婪搜索,用粒子群优化算法代替贪婪搜索就可以降低搜索算法的复杂度。

把遗传算法用于稀疏分解最优原子搜索的算法^[9]和本文算法类似,不同之处在于遗传算法搜索空间是离散化的参数空间,本文所用的粒子群优化算法搜索的参数空间是连续的,因此,本文参数空间决定的超完备字典中的“词汇量”更大,搜索结果更符合信号特征,稀疏分解在更少的步骤内收敛,相应地,对信号的代表更稀疏。

参考文献:

- [1] Mallat S, Zhang Z. Matching Pursuits with Time-frequency Dictionaries [J]. IEEE Trans. Signal Process, 1993, 41(12): 3397-3415.
- [2] Tropp J A. Topics in Sparse Approximations [D]. Ph.D. Dissertation, Computational and Applied Mathematics, the University of Texas at Austin, August 2004.
- [3] Chen S, Donoho D, Saunders M. Atomic Decomposition by Basis Pursuit [J]. SIAM Journal on Scientific Computing, 1999, 20: 33-61.
- [4] Kennedy J, Eberhart R. Particle Swarm Optimization [C]//IEEE International Conference on Neural Networks (Perth, Australia), IEEE Service Center, Piscataway, NJ, 1995, IV: 1942-1948.
- [5] 霍霄华,陈岩,朱华勇,等.多UCAV协同控制中的任务分配模型及算法[J].国防科技大学学报,2006(3):83-88.
- [6] Arthur P L, Philipos CL. Voiced/Unvoiced Speech Discrimination in Noise Using Gabor Atomic Decomposition [C]//Proc. of IEEE ICASSP, Hong Kong: IEEE Press, 2003, I(4): 820-828.
- [7] Peotta L, Granai L, Vandergheynst P. Very Low Bit Rate Image Coding Using Redundant Dictionaries [J]. SPIE, 2003, 5207: 228-239.
- [8] Elad M. Sparse Representations and the Basis Pursuit Algorithm [J]. http://www.cs.technion.ac.il/~elad/taks/2003/BP__IPAM.ppt.
- [9] 尹忠科,王建英, Vandergheynst P. 基于GA和原子特性的信号稀疏分解[J].铁道学报,2005,27(3):58-61.