采用遗传算法的分层贪婪字典训练算法

徐健[1,2]，景明利[1]，齐春[1]，常志国[3]
（1.西安交通大学电子与信息工程学院，710049，西安；2.西安邮电学院通信与信息工程学院，710121，西安；
3.长安大学信息工程学院，710064，西安）

摘要：针对稀疏表示残差过大的问题，提出了采用遗传算法的分层贪婪字典训练算法. 该算法首先将数据样本变成一维信号，然后将问题划分为若干个子问题，采用贪婪算法思想分层训练字典. 为了以一定概率寻找到每一层字典的最优值，使用遗传算法来训练每一层字典，最后将每层字典级联作为最终的字典. 在训练每一层字典时，先采用号码矩阵对样本的分类进行表示，然后以平均低秩逼近的残差能量作为衡量适应度的参数，以联赛选择的方式选出优胜个体，通过单点交叉和变异方法产生新的个体. 对二值序列的稀疏表示信号重建的实验结果表明，该算法在训练样本量较小的情况下，与传统的核奇异值分解算法相比，训练得到的字典在同样的稀疏度约束下重建信噪比提高了10倍以上.

关键词：稀疏表示；信号重建；字典训练；遗传算法
中图分类号：TP391 **文献标志码**：A **文章编号**：0253-987X(2012)04-0018-06

A Greedy Layer-Wise Dictionary Training Algorithm Based on Genetic Approach

XU Jian[1,2], JING Mingli[1], QI Chun[1], CHANG Zhiguo[3]
(1. School of Electronics and Information Engineering, Xi′an Jiaotong University, Xi′an 710049, China; 2. School of Communication and Information Engineering, Xi′an University of Posts & Telecommunications, Xi′an 710121, China; 3. School of Information Engineering, Chang′an University, Xi′an 710064, China)

Abstract: A greedy layer-wise dictionary training algorithm based on genetic approach is presented to solve the problem of too large residual in sparse representation. The algorithm firstly changes the data samples to one-dimension vectors. Then the greedy layer-wise dictionary training algorithm is employed to separate the dictionary training problem into several sub-problems. When the proposed algorithm is used to train every dictionary layer, and a genetic approach is used to find the optimal solution in every dictionary layer with a high probability. Finally, each dictionary layer is connected to get the final dictionary. When one dictionary layer is trained, matrices with numbers are used to describe the classes. Then, the average residual energy of low rank approximations is used as a measure of fitness, and Winners are selected by matching. New individuals are generated by single point crossover and mutation. Experiments about the sparse representation of the short binary sequences show that the reconstruction SNR of the proposed algorithm is 10 or more times higher than that of traditional kernel singular value decomposition algorithms under the same sparsity constraint when the number of training data samples is small.

Keywords: sparse representation; signal reconstruction; dictionary training; genetic algorithm

---

收稿日期：2011-08-11. 作者简介：徐健(1981—)，女，博士生；齐春(通信作者)，男，教授，博士生导师. 基金项目：国家自然科学基金资助项目(60972124)；中央高校基本科研业务专项基金资助项目(CHD2012JC012).
网络出版时间：2012-01-09 网络出版地址：http://www.cnki.net/kcms/detail/61.1069.T.T20120109.2052.002.html

图像稀疏表示是近年来图像处理和模式识别领域的研究热点，其中稀疏表示的字典训练是稀疏表示领域的重要研究分支.

目前关于稀疏表示字典训练的方法有 2 类. 一类是事先定义好字典数学表达式的稀疏表示方法，例如傅里叶变换(DFT)、小波变换(DWT)、曲波(Curvelets)变换 、轮廓波(Contourlet)变换等[1]，由于数学处理比较方便，这些变换都拥有快速算法，因此被广泛应用于图像处理与模式识别领域[2]，但这些算法对图像的几何特征过于依赖，对于拥有多种几何特征的自然图像，事先定义数学表达式的稀疏表示方法并不理想. 另一类是基于学习的表示方法[3-7]，如：核奇异值分解(kernel stngular value decomposition, KSVD)[3] 算法的主要思想是利用 SVD 的最佳逼近，在迭代过程中同时更新字典列和稀疏表示系数，但该方法在字典训练过程中要求远远大于数据维数的样本参与训练(例如在训练 64×256 维字典时使用了 4 万个训练样本)，否则算法将会失效；Lee 等人[4] 提出了一种快速的计算稀疏表示系数的方法，大大提高了稀疏字典的训练速度；Marial 等人[5] 提出了串行的字典训练算法，该算法的特点是训练样本可以单个参与训练，保证了字典更新的实时性，但该算法稀疏表示残差较大；Marial 等人[6] 提出了多尺度的字典训练算法，该算法将字典训练目标分为不同尺度进行训练，能够得到不同尺度的信号特征；Rubinstein 等人[7] 的字典训练方法可以同时使字典和系数稀疏，并兼顾了字典的适应性和运算效率. 基于学习的表示方法已经被证明在图像修补、去噪、复原等领域有很好的应用效果[8-10].

理论证明，能够以最低的稀疏度表示一组信号的字典和稀疏表示系数是唯一的[11]，但是上述的字典训练算法并没有证明其训练结果就是能够以最低稀疏度表示一组样本的唯一字典.

由于遗传算法在解决 NP 难问题上有许多优势，能够以一定概率搜索到全局最优解，因此它通常用于非凸问题的全局最优值的搜索.

遗传算法的主要思想是仿照生物界的“适者生存”的规律，即最适合环境的群体往往产生更多的后代，这些后代将遗传这些群体的基因. 经过多代遗传，适应度低的基因将被淘汰，最终留下适应度高的基因，即优化结果.

工程问题所使用的遗传算法往往将搜索空间映射为遗传空间，将问题的解用一种编码来表示. 遗传算法包括选择、交叉和变异 3 个主要步骤. 选择是优胜劣汰的过程，即将当前个体计算其适应度，然后以一定的选择方法决定个体的保留和淘汰；交叉是将选择得到的父代个体的部分加以替换，重组产生新的个体；变异是对群体中个体串上的某些基因位置上的基因值作改变[12].

本文提出了一种全新的采用遗传算法的字典训练算法(genetic dictionary training algorithm, GDT)算法. 该算法使用贪婪分层结构训练字典，每层字典都使用遗传算法训练，可以以较大概率获得每层字典的全局最优值. 本文还证明了 GDT 算法的收敛性. 实验表明，在训练样本较少的情况下，本文算法仍然能够训练出重建信噪比较高的稀疏表示字典.

1　分层贪婪字典训练算法

1.1　二维数据转化为一维的技巧

假设 $\boldsymbol{Y}=[\boldsymbol{y}_1,\boldsymbol{y}_2,\cdots,\boldsymbol{y}_n]\in\mathbf{R}_{m\times n}$ 表示一组待稀疏表示的样本，$\boldsymbol{Y}$ 的每一列代表一个样本，$\boldsymbol{D}$ 为字典，$\boldsymbol{X}=[\boldsymbol{x}_1,\boldsymbol{x}_2,\cdots,\boldsymbol{x}_n]\in\mathbf{R}_{l\times n}$ 为每个样本的稀疏表示系数，$\|\cdot\|_{\mathrm{F}}$ 表示矩阵的 F 范数. 字典训练模型可以表示为

$$\min_{\boldsymbol{D},\boldsymbol{X}}\|\boldsymbol{Y}-\boldsymbol{D}\boldsymbol{X}\|_{\mathrm{F}}^{2},\quad \|\boldsymbol{x}_i\|_0\leqslant T \tag{1}$$

如果处理的是图像样本，图像矩阵应变成 1 维. 为了提高样本的连续性，图像变成 1 列的方法如下式所示

$$\boldsymbol{A}=\begin{bmatrix}a_{11}&a_{12}&a_{13}\\a_{21}&a_{22}&a_{23}\\a_{31}&a_{32}&a_{33}\end{bmatrix}\rightarrow\boldsymbol{A}'=\begin{bmatrix}a_{11}\\a_{21}\\a_{31}\\a_{32}\\a_{22}\\a_{12}\\a_{13}\\a_{23}\\a_{33}\end{bmatrix} \tag{2}$$

1.2　分层贪婪字典训练算法原理

下面采用分层训练的思想训练字典. $\boldsymbol{D}^1$ 表示第 1 层字典，$\boldsymbol{d}_i^1$ 表示 $\boldsymbol{D}^1$ 的第 i 列，$\boldsymbol{X}^1$ 表示信号在第 1 层字典下得到的稀疏表示系数，$\boldsymbol{x}_i^1$ 表示 $\boldsymbol{X}^1$ 的第 i 列. 假设 $\boldsymbol{D}^1=[\boldsymbol{d}_1^1,\boldsymbol{d}_2^1,\cdots,\boldsymbol{d}_{c_1}^1]\in\mathbf{R}_{m\times c_1}$，$\boldsymbol{X}^1=[\boldsymbol{x}_1^1,\boldsymbol{x}_2^1,\cdots,\boldsymbol{x}_n^1]\in\mathbf{R}_{c_1\times n}$，$c_1$ 是 $\boldsymbol{D}^1$ 的列数，那么第 1 层字典的训练模型可以表示为

$$\min_{D^1,\boldsymbol{x}^1}\|\boldsymbol{Y}-\boldsymbol{D}^1\boldsymbol{X}^1\|_{\mathrm{F}}^{2},\quad \|\boldsymbol{x}_i^1\|_0=1 \tag{3}$$

式中：$\| \cdot \|_0$ 表示0范数，即向量中不为0的元素个数. 由式(3)可得

$$\| \boldsymbol{x}_1^1 \|_0 = \| \boldsymbol{x}_2^1 \|_0 = \cdots = \| \boldsymbol{x}_n^1 \|_0 = 1 \quad (4)$$

式(4)即表明要对 $\boldsymbol{y}_1, \boldsymbol{y}_2, \cdots, \boldsymbol{y}_n$ 进行聚类，聚类的条件是每一类可以以同一原子进行稀疏度为1的稀疏表示. 图1为二维情况下进行稀疏度为1的稀疏表示时样本和原子方向之间的关系. 如果这一过程通过SVD与遗传算法相结合的算法进行，就能以较大的概率避免陷入局部极小值点，这个算法将在第2部分详细介绍.

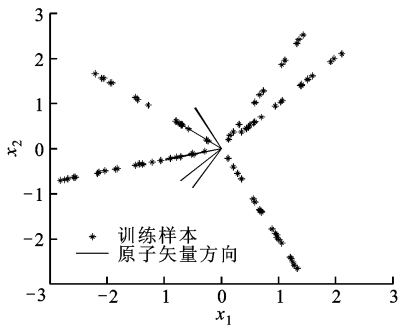

图1 二维信号的稀疏表示示意图

训练完第1层原子后，该字典的稀疏表示残差为

$$\boldsymbol{\varepsilon} = \boldsymbol{Y} - \boldsymbol{D}^1 \boldsymbol{X}^2 \quad (5)$$

将 $\boldsymbol{\varepsilon}$ 作为下一层原子的训练样本，令 $\boldsymbol{Y}' = \boldsymbol{\varepsilon}$，再以同样的方法进行第2层字典 $\boldsymbol{D}^2$ 的训练. 以此类推，训练到第 k 层原子，直到达到满足信噪比条件为止. 最终的稀疏表示字典为

$$\boldsymbol{D} = [\boldsymbol{D}^1, \boldsymbol{D}^2, \cdots, \boldsymbol{D}^k] \quad (6)$$

本文借鉴了工程优化中贪婪算法的思想，试图对字典进行分层优化，尽管不能保证每一层最优的字典合起来一定是全局最优字典，但是也能够寻找到性能较优的字典.

2 GDT算法

2.1 问题描述

假设 $\boldsymbol{T}$ 是采集得到的数据，$\boldsymbol{D}$ 是某一层字典，$\boldsymbol{X}$ 是稀疏表示系数，$\boldsymbol{x}_i$ 是 $\boldsymbol{X}$ 的第 i 列. 针对每一层字典及系数的双变量优化问题都可以表示为

$$\min_{\boldsymbol{D},\boldsymbol{X}} \| \boldsymbol{Y} - \boldsymbol{D}\boldsymbol{X} \|_F^2, \quad \| \boldsymbol{x}_i \|_0 = 1 \quad (7)$$

该优化问题带有0范数约束. 0范数约束是非凸约束，因此解决该问题不能采用传统的凸规划方法，本文设计了GDT算法来解决这个问题.

2.2 算法原理

定理1[13] 矩阵 $\boldsymbol{A}$ 的秩 k 近似记为 $\boldsymbol{A}_k$，$k < r = \mathrm{rank}(\boldsymbol{A})$，$\mathrm{rank}(\cdot)$ 表示求矩阵的秩. 矩阵 $\boldsymbol{A}_k$ 定义为 $\boldsymbol{A}_k = \sum_{i=1}^{k} \sigma_i \boldsymbol{u}_i \boldsymbol{v}_i^{\mathrm{H}}$，$k < r$，则 $\boldsymbol{A}$ 与任一矩阵 $\boldsymbol{B}$ 之差的F范数为

$$\min_{\mathrm{rank}(B)=k} \| \boldsymbol{A} - \boldsymbol{B} \|_F^2 = \| \boldsymbol{A} - \boldsymbol{A}_k \|_F^2 = \sigma_{k+1}^2 + \sigma_{k+2}^2 + \cdots + \sigma_r^2 \quad (8)$$

式中：$\boldsymbol{A} = \boldsymbol{U}\boldsymbol{S}\boldsymbol{V}^{\mathrm{H}}$ 为 $\boldsymbol{A}$ 的奇异值分解，$\boldsymbol{U} = [\boldsymbol{u}_1, \boldsymbol{u}_2, \cdots, \boldsymbol{u}_{m_1}]$，$\boldsymbol{V} = [\boldsymbol{v}_1, \boldsymbol{v}_2, \cdots, \boldsymbol{v}_{m_2}]$，$m_1$ 为 $\boldsymbol{U}$ 的列数，m_2 为 $\boldsymbol{V}$ 的列数，$\boldsymbol{S}$ 为对角阵，对角线上的元素为奇异值 $\sigma_1 \geqslant \sigma_2 \geqslant \cdots \geqslant \sigma_r$.

由定理1可得，$\boldsymbol{A}_1 = \sigma_1 \boldsymbol{u}_1 \boldsymbol{v}_1^{\mathrm{H}}$ 是秩为1的 $\boldsymbol{A}$ 的最佳逼近矩阵.

2.3 遗传字典训练算法

2.3.1 *初始种群的产生* 给 $\boldsymbol{Y} = [\boldsymbol{y}_1, \boldsymbol{y}_2, \cdots, \boldsymbol{y}_n]$ 编号，例如 $\boldsymbol{y}_1$ 编为1号，$\boldsymbol{y}_n$ 编为 n 号等. 初始种群的产生方法为：需要分 c_1 个类别，令每个号码随机选择自己所属的类别，以矩阵的形式标记. 例如：当 $n = 10$ 时，需要分3个类别. 随机产生的其中一个标记矩阵为

$$\boldsymbol{C}_j = \begin{bmatrix} 1 & 7 & 8 & 9 \\ 6 & 2 & 3 & 10 \\ 4 & 5 & 0 & 0 \end{bmatrix} \quad (9)$$

该矩阵表示1、7、8、9号样本属于第1类，6、2、3、10号样本属于第2类，4、5号样本属于第3类，其他没有分配样本的矩阵元素用0填充. 通过这种方法可以产生 β 个个体 $\{\boldsymbol{C}_1, \boldsymbol{C}_2, \cdots, \boldsymbol{C}_\beta\}$. 为了提高种群的多样性，每个个体矩阵不都是同样的行数，可以将行数限制在一定范围内. 但是，矩阵中的元素要保证每个样本号码都要出现，且不重复.

2.3.2 *计算适应值* 取每个个体中的每一行对应号码的样本组成一个矩阵，进行SVD. 如式(9)所表示的个体，可以构成3个矩阵

$$\left.\begin{aligned} \boldsymbol{Y}_1 &= [\boldsymbol{y}_1, \boldsymbol{y}_7, \boldsymbol{y}_8, \boldsymbol{y}_9] \\ \boldsymbol{Y}_2 &= [\boldsymbol{y}_6, \boldsymbol{y}_2, \boldsymbol{y}_3, \boldsymbol{y}_{10}] \\ \boldsymbol{Y}_3 &= [\boldsymbol{y}_4, \boldsymbol{y}_5] \end{aligned}\right\} \quad (10)$$

分别对 $\boldsymbol{Y}_1$、$\boldsymbol{Y}_2$、$\boldsymbol{Y}_3$ 进行SVD，即 $\boldsymbol{Y}_1 = \boldsymbol{U}_1\boldsymbol{S}_1\boldsymbol{V}_1^{\mathrm{H}}$，$\boldsymbol{Y}_2 = \boldsymbol{U}_2\boldsymbol{S}_2\boldsymbol{V}_2^{\mathrm{H}}$，$\boldsymbol{Y}_3 = \boldsymbol{U}_3\boldsymbol{S}_3\boldsymbol{V}_3^{\mathrm{H}}$，取 $\boldsymbol{U}_1$ 的第1列 $\boldsymbol{u}_{1,1}$，$\boldsymbol{S}_1$ 的第1行第1列的元素 $\sigma_{1,1}$，$\boldsymbol{V}_1$ 的第2列 $\boldsymbol{v}_{1,1}$，令 $\boldsymbol{Y}_1$ 的秩为1的近似矩阵 $\widetilde{\boldsymbol{Y}}_1 = \boldsymbol{u}_{1,1}\boldsymbol{\sigma}_{1,1}\boldsymbol{v}_{1,1}$，根据定理1，$\widetilde{\boldsymbol{Y}}_1$ 是 $\boldsymbol{Y}_1$ 的秩为1的最佳逼近. 同样的方法计算出 $\widetilde{\boldsymbol{Y}}_2$、$\widetilde{\boldsymbol{Y}}_3$. 计算残差能量 $\boldsymbol{E}_1 = \| \boldsymbol{Y}_1 - \widetilde{\boldsymbol{Y}}_1 \|_F^2$，同样的方法求 $\boldsymbol{E}_2$、$\boldsymbol{E}_3$. 计算第 j 个个体的信噪比

$$R_{\mathrm{SN},j} = 10\lg[\| \boldsymbol{Y} \|_F^2 / (\boldsymbol{E}_1 + \boldsymbol{E}_2 + \boldsymbol{E}_3)] \quad (11)$$

将信噪比作为适应值,种群中每个个体都计算一个适应值,得到$\{R_{\mathrm{SN},1},R_{\mathrm{SN},2},\cdots,R_{\mathrm{SN},\beta}\}$.

2.3.3　选择策略　为了便于计算,在每次选择交叉变异之后都固定种群中个体与初始种群的数目相同. 例如设置选择率为 0.8,那么 β 个个体中将有 $\beta\times80\%$个优秀个体被选中,选择采用联赛方式进行.

联赛方式为每次随机选择 N 个个体,组成联赛队员,评比其适应值,选出 b 个最优的个体. 这样共选择 M 次,直到选中的个体数目符合要求即可. 实验证明,这类选择方式比普通的轮盘赌选择方式更能避免陷入局部极小点.

2.3.4　交叉和变异　在选择完成后的个体中随机选取一部分作为交叉对象,以中点作为交叉点. 例如随机选择了第 λ 代的 2 个个体 $\boldsymbol{C}_i^{\lambda}$ 和 $\boldsymbol{C}_j^{\lambda}$,交叉产生第 $\lambda+1$ 代的个体 $\boldsymbol{C}_i^{\lambda+1}$ 和 $\boldsymbol{C}_j^{\lambda+1}$,即

$$\left.\begin{aligned}\boldsymbol{C}_i^{\lambda}&=\begin{bmatrix}1&7&8&9\\6&2&3&10\\4&5&0&0\end{bmatrix}\\ \boldsymbol{C}_j^{\lambda}&=\begin{bmatrix}1&2&6&8&0&0&0&0\\9&7&3&4&5&10&0&0\end{bmatrix}\end{aligned}\right\}\tag{12}$$

为了保证每个样本都能够得到分类,需要对每个个体进行去掉重复号码和补充缺少号码的操作. 交叉并去重补缺后的个体为

$$\left.\begin{aligned}\boldsymbol{C}_i^{\lambda+1}&=\begin{bmatrix}1&7&8&9\\6&2&0&0\\3&4&5&10\end{bmatrix}\\ \boldsymbol{C}_j^{\lambda+1}&=\begin{bmatrix}1&2&6&8&7&9\\4&5&3&10&0&0\end{bmatrix}\end{aligned}\right\}\tag{13}$$

变异方法是将个体中的部分元素换个位置,例如

$$\begin{bmatrix}1&7&8&9\\6&2&0&0\\3&4&5&10\end{bmatrix}\text{变异为}\begin{bmatrix}1&7&8&0\\6&2&5&9\\3&4&10&0\end{bmatrix}\tag{14}$$

与其他遗传算法过程相同,经过选择交叉变异后得到的种群再次进行下一代的遗传.

3　算法收敛性分析

根据模式理论,假设矩阵中的任一模式为[14]

$$\boldsymbol{H}=\begin{bmatrix}a_1&a_2&\cdots&a_m&\delta&\cdots&\delta\\ \delta&\delta&\cdots&\delta&\delta&\cdots&\delta\\ \delta&\delta&\cdots&\delta&\delta&\cdots&\delta\end{bmatrix}\tag{15}$$

式中:δ 表示任意的实数. 为了方便分析作如下假设:

(1)假设种群中每个代表的个体矩阵都是同样的行数;

(2)假设在杂交过程中,补缺不会对关注的模式 $\boldsymbol{H}$ 的个体数产生影响;

(3)由于在一般遗传算法中,变异率都会设很低的值,因此可忽略变异对模式 $\boldsymbol{H}$ 个体数的影响;

(4)假设群体中所有个体都是同样的行数,即规定每个个体矩阵都将样本分成同样多的类别.

设 $m=m(\boldsymbol{H},t)$ 为模式 $\boldsymbol{H}$ 在第 t 代群体中所含的个数,含有 $\boldsymbol{H}$ 的个体串为$\{\boldsymbol{w}_1,\boldsymbol{w}_2,\cdots,\boldsymbol{w}_m\}$,则模式 $\boldsymbol{H}$ 在第 t 代群体中的平均适应值为

$$f(\boldsymbol{H},t)=\sum_{j=1}^{m}f(\boldsymbol{w}_j)/m\tag{16}$$

那么,第 t 代整个群体为 $\boldsymbol{P}_t=\{\boldsymbol{w}_1,\boldsymbol{w}_2,\cdots,\boldsymbol{w}_n\}$. 个体以概率 $p_i=f(w_i)/\sum_{i=1}^{n}f(\boldsymbol{w}_i)$ 被选择,则下一代通过选择得到模式 $\boldsymbol{H}$ 的生存数量为

$$m_{\mathrm{c}}(\boldsymbol{H},t+1)=m(\boldsymbol{H},t)nf(\boldsymbol{H},t)/\sum_{i=1}^{n}f(w_i)\tag{17}$$

将群体的平均适应值表示为 $\overline{f}=\sum_{i=1}^{n}f(\boldsymbol{w}_i)/n$,则选择后的模式 $\boldsymbol{H}$ 的生存数量为

$$m_{\mathrm{c}}(\boldsymbol{H},t+1)=m(\boldsymbol{H},t)f(\boldsymbol{H},t)/\overline{f}\tag{18}$$

交叉的对象为选择后的个体,假设群体的杂交率为 p_{cr},应对 $p_{\mathrm{cr}}n$ 个个体进行交叉. 假设通过选择后下一代个体总数为 k,每次选择到的 2 个杂交对象都含有模式 $\boldsymbol{H}$ 的概率为$\left(\frac{m_{\mathrm{c}}(\boldsymbol{H},t+1)}{k}\right)^2$,这 2 个对象依照上一节提到的交叉方法不会破坏模式 $\boldsymbol{H}$. 选到一个含有模式 $\boldsymbol{H}$、另一个不含模式 $\boldsymbol{H}$ 的概率为$\frac{m_{\mathrm{c}}(\boldsymbol{H},t+1)}{k}\left(1-\frac{m_{\mathrm{c}}(\boldsymbol{H},t+1)}{k}\right)$,交叉但未淘汰相同元素时,其中一个含有模式 $\boldsymbol{H}$,但是淘汰的过程可能会破坏模式 $\boldsymbol{H}$,假设淘汰过程不破坏 $\boldsymbol{H}$ 的概率为 p_{d}.

假设那个不含有模式 $\boldsymbol{H}$ 的个体的所有元素位于下半段的概率为 1/2,所以$\{a_1,a_2,\cdots,a_m\}$中每个元素处于下半段的概率均为 1/2. 任意 b 个元素处于下半段,另外 $m\sim b$ 个元素都处于上半段的概率为 $\boldsymbol{C}_m^b(1/2)^b(1/2)^{m-b}$,其中 $\boldsymbol{C}_m^b$ 表示 m 个元素的 b 种排列的排列数. 每对重复元素以相等的概率决定谁被淘汰,又因为淘汰的全是下半段才能保证 $\boldsymbol{H}$ 不被破坏,所以上述情况 $\boldsymbol{H}$ 不被破坏的概率为 $\boldsymbol{C}_m^b(1/2)^b(1/2)^{m-b}(1/2)^b$,所以有

$$p_{\mathrm{d}}=\boldsymbol{C}_m^1\left(\frac{1}{2}\right)^1\left(\frac{1}{2}\right)^{m-1}\left(\frac{1}{2}\right)^1+$$

$$C_m^2\left(\frac{1}{2}\right)^2\left(\frac{1}{2}\right)^{m-2}\left(\frac{1}{2}\right)^2+\cdots+C_m^m\left(\frac{1}{2}\right)^{2m} \tag{19}$$

则通过选择得到的 $\boldsymbol{H}$ 的个体数量为

$$m_{\text{cross}}(\boldsymbol{H},t+1)=\left(2\left(\frac{m_{\text{c}}(\boldsymbol{H},t+1)}{k}\right)^2+\frac{m_{\text{c}}(\boldsymbol{H},t+1)}{k}\left(1-\frac{m_{\text{c}}(\boldsymbol{H},t+1)}{k}\right)p_{\text{d}}\right)p_{\text{cr}}n \tag{20}$$

根据前面的假设，忽略得到的含 $\boldsymbol{H}$ 的个体数目，因此有

$$m(\boldsymbol{H}+1,t)=m_{\text{c}}(\boldsymbol{H},t+1)+m_{\text{cross}}(\boldsymbol{H},t+1)=\left[2\left(\frac{m(\boldsymbol{H},k)}{k}\right)^2-\left(\frac{m(\boldsymbol{H},k)}{k}\right)^2p_{\text{d}}\right]\left(\frac{f(\boldsymbol{H})}{\bar{f}}\right)^2p_{\text{cr}}n+\left(m(\boldsymbol{H},k)+\frac{m(\boldsymbol{H},k)}{k}p_{\text{d}}p_{\text{cr}}n\right)\frac{f(\boldsymbol{H})}{\bar{f}} \tag{21}$$

从式(21)可以看出，当模式的适应值大于平均适应值时，随着迭代次数的增加，拥有该模式的个体数量将会增加.

4 实 验

为了提高实验的效率，并且模拟图像信号的整数特性，本文通过穷举二值5维信号产生了一组样本信号，该信号只有{0,1}2个元素组成，每个信号的长度为5. 穷举这类信号的所有32种状态，由此产生的样本信号矩阵为

$$\boldsymbol{Y}=\begin{bmatrix}0&0&0&\cdots&1\\0&0&0&\cdots&1\\0&0&0&\cdots&1\\0&0&1&\cdots&1\\0&1&0&\cdots&1\end{bmatrix} \tag{22}$$

式中：$\boldsymbol{Y}\in\mathbf{R}_{5\times32}$. 假设我们能够训练一个字典 $\boldsymbol{D}$ 和表示系数 $\boldsymbol{X}=[\boldsymbol{x}_1,\boldsymbol{x}_2,\cdots,\boldsymbol{x}_{32}]$，满足

$$\min_{D,X}\|\boldsymbol{Y}-\boldsymbol{D}\boldsymbol{X}\|_{\text{F}}^2,\quad\|\boldsymbol{x}_i\|_0\leqslant T,\quad i=1,2,\cdots,32 \tag{23}$$

那么，意味着该字典可以以稀疏度 T 稀疏表示这类信号的所有状态.

稀疏表示的信噪比定义为

$$R_{\text{SN}}=10\lg\left(\frac{\|\boldsymbol{Y}\|_{\text{F}}^2}{\|\boldsymbol{Y}-\boldsymbol{D}\boldsymbol{X}\|_{\text{F}}^2}\right) \tag{24}$$

其中字典规模使用矩阵表示，例如[5,6,2]表示训练结果该字典总共有13列，第1轮训练了5列，第2轮训练了6列，第3轮训练了2列. 训练该字典时，本文统一设置变异率为0. KSVD和本文算法的二值短序列稀疏重构实验结果如表1所示. 从表1的实验结果来看，GDT算法得到的稀疏表示信噪比远远高于KSVD算法.

表1 2种算法二值短序列稀疏重构的实验结果

字典总列数	字典分配	种群规模	杂交率	稀疏度	信噪比/dB	
					GDT算法	KSVD算法
17	[11,6]	600	0.2	2	310	25
14	[7,7]	600	0.3	2	312	27

5 结 论

本文提出了一种基于遗传算法的字典训练算法，该算法利用分层贪婪算法的思想寻找优化每一层字典. 二值短序列的实验结果显示，该算法所得的到字典重构信噪比与经典KSVD算法相比提高了10倍以上. 本文算法运算量较大，减少算法计算复杂度及对于复杂信号与图像的字典训练应用是进一步的研究方向. 本文算法采用贪婪算法进行求解，贪婪算法在对问题求解时，总是做出在当前看来是最好的选择. 因此，在训练每一层字典时通过遗传算法能够以较大概率得到式(7)所示优化模型的最优值，即每一层字典的最优值. 但是，贪婪算法不是对问题都能得到整体最优解，对范围相当广泛的许多问题只能得到整体最优解的近似解即次优解，本算法即使得到式(7)所示的全局最优值，也不能保证得到式(1)所示模型的全局最优性. 分层贪婪字典训练算法在得到每一层字典最优的情况下，所获得的整体字典在大多数情况下为次优的字典，因此如何寻找全局最优字典仍然是这一类课题有待解决的问题.

参考文献：

[1] RUBINSTEIN R, BRUCKSTEIN A M, ELAD M. Dictionaries for sparse representation modeling [J]. Proceedings of the IEEE, 2010, 98(6): 1045-1057.

[2] CHEN S, DONOHO D, SAUNDERS M. Atomic decomposition by basis pursuit [J]. SIAM Journal on Scientific Computing, 1999, 20(1): 33-61.

[3] AHARON M, ELAD M, BRUCKTEIN A. K-SVD: an algorithm for designing overcomplete dictionaries for sparse representation [J]. IEEE Transactions on Signal Processing, 2006, 54(11): 4311-4322.

[4] LEE H, BATTLE A, RAINA R, et al. Efficient sparse coding algorithms [J]. Advances in Neural Information Processing Systems, 2007, 19(1), 801-808.

[5] MAIRAL J, BACH F, PONCE J, et al. Online dictionary learning for sparse coding [C]// Proceeding of

ICML'09. Montreal, Canada: Association for Computing Machinery, 2009: 689-696.

[6] MAIRAL J, SAPIRO G, ELAD M. Multiscale sparse image representation with learned dictionaries [C]// Proceedings of ICIP'07. San Antonio, TX, USA: Society for Industrial and Applied Mathematics Publications, 2007: 214-241.

[7] RUBINSTEIN R, ZIBULEVSKY M, ELAD M. Double sparsity: learning sparse dictionaries for sparse signal approximation [J]. IEEE Transactions of Signal Processing, 2010, 58(3): 1553-1564.

[8] XU Zongben, SUN Jian. Image inpainting by patch propagation using patch sparsity [J]. IEEE Transactions on Image Processing, 2010, 19(5): 1153-1165.

[9] 王国栋，阳建宏，黎敏，等. 基于自适应稀疏表示的宽带噪声去除算法 [J]. 仪器仪表学报，2011，32(8)：1818-1823.
WANG Guodong, YANG Jianhong, LI Min, et al. Wideband noise removing algorithm based on adaptive sparse representation [J]. Chinese Journal of Scientific Instrument, 2011, 32(8): 1818-1823.

[10] 冯亮，王平，许廷发，等. 运动模糊退化图像的双字典稀疏复原 [J]. 光学精密工程，2011，19(8)：1982-1989.
FENG Lang, WANG Ping, XU Tingfa, et al. Dual dictionary sparse restoration of blurred images [J]. Optics and Precision Engineering, 2011, 19(8): 1982-1989.

[11] AHARON M, ELAD M, BRUCKTEIN A. On the uniqueness of overcomplete dictionaries, and a practical way to retrieve them [J]. Linear Algebra and Its Applications, 2006, 416(1): 48-67.

[12] 张可村，李换琴. 工程优化方法及应用 [M]. 西安：西安交通大学出版社，2007：238-246.

[13] 张贤达. 矩阵分析与应用 [M]. 北京：清华大学出版社，2004：352-358.

[14] 李敏强，寇纪淞. 遗传算法的基本理论与应用 [M]. 北京：科学出版社，2002：120-133.

（编辑　刘杨）