

基于学习字典的机器人图像稀疏表示方法

郭俊锋¹ 李育亮¹

摘要 针对机器人图像压缩感知 (Compressed sensing, CS) 过程中稀疏字典训练时间过长的问题, 本文提出了一种更加高效的字典学习方法. 通过对 MOD、K-SVD、SGK 等字典学习算法研究, 从参与更新的字典原子列数入手, 将残差项变形为多列原子同时更新, 进而利用最小二乘法连续地更新字典中的多个原子. 本文算法是对 SGK 算法字典学习效率的进一步提高, 减少了单次迭代的计算量, 加快了字典学习速度. 实验表明, 本文算法与 K-SVD 和 SGK 算法相比, 在字典稀疏性和重构图像质量变化很小的情况下, 字典训练时间得到较明显缩短.

关键词 压缩感知, 稀疏表示, 字典学习, 最小二乘法

引用格式 郭俊锋, 李育亮. 基于学习字典的机器人图像稀疏表示方法. 自动化学报, 2020, 46(4): 820–830

DOI 10.16383/j.aas.2018.c170352

Sparse Representation of Robot Image Based on Dictionary Learning Algorithm

GUO Jun-Feng¹ LI Yu-Liang¹

Abstract Aiming at the problem that the time of dictionary training is too long in the process of robot image compressed sensing (CS), a more efficient dictionary learning algorithm is proposed in this paper. Through the research of MOD, K-SVD, SGK and other dictionary learning algorithms, we start with the number of dictionary atoms involved in the update, and change the residuals into multiple atoms updating at the same time. Then, we can update the multiple atoms of the dictionary sequentially by using the least square method. The algorithm is to further improve the efficiency of the SGK algorithm, and to reduce the computation of single iteration. The experiment results show that the improved algorithm can effectively shorten the dictionary training time under the premise of ensuring sparsity of the dictionary and the quality of the reconstructed image.

Key words Compressed sensing (CS), sparse representation, dictionary learning, the least squares method

Citation Guo Jun-Feng, Li Yu-Liang. Sparse representation of robot image based on dictionary learning algorithm. *Acta Automatica Sinica*, 2020, 46(4): 820–830

智能机器人通过多种传感器与外界交流, 据研究表明, 其中约 70% 的外部信息来自视觉图像. 传统的视觉图像采样是基于采样频率不能低于模拟信号频谱中最高频率的 2 倍的香农采样定理, 这种采样方式将会得到巨量的数据, 后期处理工作量大速度慢, 导致机器人运行迟钝行为缓慢. 近些年来, Candes 提出的全新的信号压缩采样方法压缩感知 (Compressed sensing, CS)^[1], 能大大降低数据的维数, 减少数据量, 其得到的测量值包含有能够重构出原图像信号的丰富信息. 对机器人视觉来说, 图像处理的主要目的是获取图像的特征信息, 并不需要完整的原始图像, 因此只须在压缩域对低维数据进行处理并获取其特征信息, 进而提高机器人的反应和

运行速度. 压缩感知的前提条件是图像信号在变换基或者说字典上具有稀疏性, 其直接影响着压缩测量数, 图像在所选择稀疏字典上越稀疏, 其所需要的压缩测量数就越少, 后期处理就越快, 反之若选择的稀疏字典不能很好地满足要求, 其数据量就越多且重构图像质量也越差. 只有获得最佳稀疏矩阵, 才能最有效并简洁地表示原始信号, 才能保证所恢复信号的精确性^[2].

信号的稀疏表示一般分为两大类方法: 第一类是分析字典, 如 DCT (Discrete cosine transform) 字典、小波字典、脊波字典、曲线波字典等; 第二类就是根据数据或者信号本身训练的过完备字典^[3]. 前者构造简单, 但字典原子形态单一不能对复杂的自然图像进行有效的稀疏表示; 而通过学习获得的冗余字典具有更好地自适应重构图像的能力, 字典中原子与待重建图像具有相关性, 能更有效表达图像局部结构特点, 使得信号在字典上有更稀疏的表示. 而且, 由于部分噪声在字典上是非稀疏的, 而图像信号在字典上具有稀疏性, 因此基于学习字典对图像进行稀疏表示具有一定去噪作用. 综上, 用学习

收稿日期 2017-06-26 录用日期 2018-03-06
Manuscript received June 26, 2017; accepted March 6, 2018
国家自然科学基金 (51465034) 资助
Support by National Natural Science Foundation of China (51465034)
本文责任编辑 黎铭
Recommended by Associate Editor LI Ming
1. 兰州理工大学机电工程学院 兰州 730050
1. School of Mechanical and Electronic Engineering, Lanzhou University of Technology, Lanzhou 730050

字典的方法对图像进行稀疏表示具有较大优势.

在权威的《自然》杂志上 Olshausen 等提出一种 Sparsenet 算法的字典学习方法^[4], 其采用梯度下降算法和最大似然估计算法获得字典训练的更新, 字典学习的理论雏形由此诞生. 受广义聚类算法的启发, Engan 等在 Sparsenet 字典学习算法的基础上提出了 MOD (Method of optimal directions) 字典学习算法^[5], 它直接采用最小二乘法解决目标函数的优化问题, 但其过程需要大矩阵的乘法和求逆计算, 对存储容量的要求高. 为了降低 MOD 算法的计算复杂度尤其是空间复杂度, Aharon 等提出了 K-SVD (K-singular value decomposition) 算法^[6], 使字典逐列更新, 提高了字典学习速度, 但其应用 SVD 分解使计算复杂度依旧较大. 王强等^[7]提出利用改进的 K-SVD 算法进行稀疏分解, 在每次奇异值分解后对多列字典原子同时赋值, 减少了迭代时间, 但信号重构质量有所下降. 易可夫等^[8]提出了一种优化字典学习算法来构造压缩数据收集中的稀疏字典, 提高了压缩数据收集对多样化传感数据的适应能力, 同时抑制环境噪声对数据收集精度的影响, 但其字典学习效率并未提高. 陆婉芸等^[9]在字典原子训练的过程中加入某种特定的排序规则, 使得各个影像字典在拥有各自影像属性的同时其原子也具备相似的排列顺序, 减小影像间差异的干扰, 但其仍沿用传统的 K-SVD 算法更新字典计算复杂度较高.

上述方法对字典更新速度上并没有大幅度提高, 计算复杂度较大, 不能适用于机器人图像的稀疏表示. 最近, 一种被称作 SGK (Sequential generalization of k-means) 的字典学习算法^[10] 被作为经典 K-SVD 算法的有效替代, 相比于 K-SVD 它具有更快的执行速度和与之相媲美的字典训练效果. 但 SGK 算法仅通过单列字典的更新, 计算复杂度较高, 时间消耗仍然较多. 为此本文提出了一种基于 SGK 算法的改进算法, 将 SGK 算法中的残差项变形为多列原子同时更新, 进而利用最小二乘法更新字典中的 r 个原子, 减少了单次迭代的计算量和稀疏表示的时间, 提高了字典学习效率.

1 字典学习算法的经典模型

MOD、K-SVD 和 SGK 算法及其基础上改进算法的思想核心都来自聚类算法, 且算法的主要步骤都包括字典的初始化、对稀疏系数矩阵 A 的稀疏编码和对字典 D 的更新这三步. 其目标函数模型可以表示为:

$$\min_{D,A} \{ \|X - DA\|_F^2 \} \quad \text{s. t.} \quad \forall_i \|\alpha_i\|_0 \leq T_0 \quad (1)$$

式中, X 表示样本信号, D 表示过完备字典, α_i 是稀疏系数矩阵 $A = [\alpha_1, \alpha_2, \dots, \alpha_N]$ 的列, $\|A\|_F$ 是

矩阵 A 的 Frobenius 范数, T_0 为用于表示信号 X 所使用的字典原子数, 即稀疏系数向量 α_i 中非零元素个数. 字典的初始化有两种选择: 一种是任意给定一个字典 (如过完备离散余弦 DCT 字典) 进行初始化; 另一种是从训练样本集中随机地选取 K 个元素. 稀疏编码阶段一般是固定初始化后的过完备字典 D , 求解稀疏系数矩阵 A . 稀疏编码阶段常用的方法有匹配追踪 (Matching pursuit, MP)、正交匹配追踪 (Orthogonal matching pursuit, OMP)、基追踪 (Basic pursuit, BP) 等. 字典更新阶段是固定稀疏编码后得到的 A , 进一步求取字典 D 的过程. 对于一组训练样本信号 $X = [\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N]$, 最优字典 D 是根据最小化表示误差 $\|X - DA\|_F^2$ 逐步迭代得到的, 因此训练字典 D 相比于分析字典将产生更好的稀疏表示效果. 各种字典学习算法的主要区别在于字典更新方法上.

MOD 算法是根据信号样本 X 与稀疏编码阶段获得的稀疏系数矩阵 A , 利用最小二乘法直接更新字典 D :

$$D^{(t)} = \arg \min_D \|X - D^{(t-1)} A^{(t)}\|_F^2 = X(A^{(t)})^T [A^{(t)}(A^{(t)})^T]^{-1} \quad (2)$$

式中, t 为迭代次数. 若 $\|X - D^{(t-1)} A^{(t)}\|_F^2$ 足够小, 则迭代停止, $D = D^{(t)}$. 由上式可以看出 MOD 算法在更新字典过程中每进行一次迭代就需要更新字典所有的列, 并进行大矩阵的乘法和求逆, 这将占用大量内存, 增加字典学习的计算复杂度.

K-SVD 算法在更新字典原子的同时也更新稀疏系数向量. 其字典更新步骤如下:

1) 先固定系数矩阵 A 与字典 D 中除待更新原子 \mathbf{d}_k 外的 $K - 1$ 个原子, 残差可表示为

$$\begin{aligned} \|X - D^{(t-1)} A^{(t)}\|_F^2 &= \left\| X - \sum_{j=1}^K \mathbf{d}_j^{(t-1)} \alpha_{rowj}^{(t)} \right\|_F^2 = \\ &= \left\| \left(X - \sum_{j \neq k} \mathbf{d}_j^{(t-1)} \alpha_{rowj}^{(t)} \right) - \mathbf{d}_k^{(t-1)} \alpha_{rowk}^{(t)} \right\|_F^2 = \\ &= \left\| E_k^{(t)} - \mathbf{d}_k^{(t-1)} \alpha_{rowk}^{(t)} \right\|_F^2 \end{aligned} \quad (3)$$

式中, $\mathbf{d}_j^{(t-1)}$ 为第 $t - 1$ 次迭代得到字典的第 j 列, $\alpha_{rowj}^{(t)}$ 为第 t 次迭代的稀疏矩阵第 j 行. 由上式可以看出, 在 K-SVD 字典更新过程中 $\sum_{j \neq k} \mathbf{d}_j^{(t-1)} \alpha_{rowj}^{(t)}$ 是固定的 $K - 1$ 项, $\mathbf{d}_k^{(t-1)} \alpha_{rowk}^{(t)}$ 是待更新项, 则 $E_k^{(t)}$ 就是除第 k 个待求解原子外的残差项. 同 MOD 一样, 极小化式 (3) 就是寻找最佳 $\mathbf{d}_k^{(t-1)}$ 和 $\alpha_{rowk}^{(t)}$ 来对 $E_k^{(t)}$ 进行逼近.

2) 为减少 SVD 分解过程中计算量, 将稀疏系数向量中非零原子所在列的标号 j 提取出来, 定义标号数组 $\omega_k = \{j \mid \alpha_{rowj}^{(t)}(j) \neq 0, 1 \leq j \leq K\}$; 再定义矩阵 Ω , 并令其元素 $(\omega_k(j), j)$ 为 1, 其他位置元素为 0. 进而, 获得 $X^* = X\omega$ 、 $E_k^{(t)*} = E_k^{(t)}\omega$ 、 $\alpha_{rowk}^{(t)*} = \alpha_{rowk}^{(t)}\omega$, 完成去零收缩.

3) 对 $E_k^{(t)*}$ 进行 SVD 分解: $E_k^{(t)*} = USV^T$, 且 U 、 V 为正交矩阵, S 为对角矩阵, 其元素非负且降序排列. 将 U 的第一列作为更新后的原子 $\mathbf{d}_k^{(t+1)}$, 矩阵 S 的 $S(1, 1)$ 乘以 V 的第一列作为更新后的稀疏行向量 $\alpha_{rowk}^{(t+1)}$. 从 K-SVD 更新过程可以看出, 字典是逐列更新的, 字典原子更新的同时稀疏系数向量又更新了一次, 虽然较 MOD 算法字典更新阶段避免了大矩阵求逆, 但总体算法时间复杂度依旧比较大, 消耗时间较多.

在字典更新步骤中 K-SVD 算法会破坏稀疏系数的结构, MOD 虽然不会破坏稀疏系数的结构, 但原子更新不是顺序更新的且计算复杂度更高^[11]. 针对这些问题, Sahoo 等提出了 SGK (Sequential generalization of k-means) 算法^[12-13], 该算法具有不破坏稀疏系数结构的优点, 不仅能够逐列更新原子, 而且算法复杂度比 K-SVD 更低^[11]. 类似于 K-mean 算法, SGK 和 K-SVD 都是通过迭代来训练字典的, 通过稀疏编码 (对 α) 和字典更新 (对 \mathbf{d}) 的交替重复直到收敛, 不同之处在于字典更新部分.

SGK 算法的字典更新阶段沿用 K-SVD 算法中固定 $K-1$ 项 $\sum_{j \neq k} \mathbf{d}_j^{(t-1)} \alpha_{rowj}^{(t)}$, 剩下的一项 $\mathbf{d}_k^{(t-1)} \alpha_{rowk}^{(t)}$ 为待求解项, 残差依旧可以写成式 (3). 虽然 MOD 能适合所有稀疏表示的应用, 在稀疏系数和字典上不计约束, 但它需要更多的计算资源去处理. 相对的, 像 K-SVD 和 K-means 这样的顺序算法可以只需较少的资源就可以完成字典更新. 因此 Sahoo 等对 K-SVD 中的模型 $\mathbf{d}_k^{(t)}, \alpha_{rowk}^{(t)} = \arg \min_{\mathbf{d}_k, \alpha_{rowk}} \left\| E_k^{(t)} - \mathbf{d}_k^{(t-1)} \alpha_{rowk}^{(t)} \right\|_F^2$ 进行修改. 如果保持 $\alpha_{rowk}^{(t)}$ 不变或者说字典更新这一步中不同时更新稀疏系数, 将不会有稀疏性损失和结构损失, 因此提出连续字典更新模型:

$$\mathbf{d}_k^{(t)} = \arg \min_{\mathbf{d}_k} \left\| E_k^{(t)} - \mathbf{d}_k^{(t-1)} \alpha_{rowk}^{(t)} \right\|_F^2 \quad (4)$$

通过与解 MOD 算法中模型如式 (2) 相同的方法 (最小二乘法), 求解式 (4) 得:

$$\mathbf{d}_k^{(t)} = E_k^{(t)} (\alpha_{rowk}^{(t)})^T \left[\alpha_{rowk}^{(t)} (\alpha_{rowk}^{(t)T}) \right]^{-1} \quad (5)$$

通过在更新序列中连续用 $\mathbf{d}_k^{(t)}$ 代替 $\mathbf{d}_k^{(t-1)}$, 这个过程依次重复所有字典原子, 类似 K-SVD 和 K-means.

但 SGK 算法仅通过单列字典的顺序更新, 运算次数依然较多, 时间消耗较大.

2 本文改进算法

本文从参与更新的字典原子列数入手, 将 SGK 算法中的残差项变形为多列原子同时更新, 减少了单次迭代的计算量, 进一步提高了字典学习效率. 算法本身依旧包含两大主要部分, 即稀疏编码和字典更新, 具体算法流程如下.

程序: 初始化 $t = 0$, 选定初始字典 $D^{(0)} \in \mathbf{R}^{n \times K}$.

1) 稀疏编码阶段: 对 $X = [\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N] \in \mathbf{R}^{n \times N}$ 稀疏编码得到稀疏矩阵 $A^{(t)}$

$$\forall_i, \alpha_i^{(t)} = \arg \min_{\alpha_i} \|\alpha_i\|_0; \|\mathbf{x}_i - D^{(t-1)} \alpha_i\|_2^2 \leq \xi \quad (6)$$

其中, ξ 表示容许误差, $A^{(t)} = [\alpha_1, \alpha_2, \dots, \alpha_N] \in \mathbf{R}^{K \times N}$, α_i 在这里指的是矩阵 $A^{(t)}$ 的列.

2) 字典更新阶段: 用稀疏编码所获得的稀疏矩阵 $A^{(t)}$ 更新字典 $D^{(t-1)}$

$$D^{(t)} = \arg \min_D \|X - D^{(t-1)} A^{(t)}\|_F^2 \quad (7)$$

并令迭代次数 $t = t + 1$.

初始化字典选用 DCT 字典, 稀疏编码方法采用 OMP 算法, 由于自然信号的稀疏性未知, 所以必须使用一个基于约束最小化的表示误差 ξ .

字典更新阶段不同于 K-SVD 和 SGK 算法中单列原子更新, 而是 r 个原子同时更新. 根据不同原子列数在稀疏分解过程中的迭代收敛次数、时间消耗与重构图像的质量, 确定最佳的字典更新的列数 r . 则残差可以写成式 (8):

$$\begin{aligned} \|X - D^{(t-1)} A^{(t)}\|_F^2 &= \left\| X - \sum_{j=1}^K \mathbf{d}_j^{(t-1)} \alpha_{rowj}^{(t)} \right\|_F^2 = \\ &= \|E_r^{(t)} - D_r^{(t-1)} A_{rowr}^{(t)}\|_F^2 \end{aligned} \quad (8)$$

其中, $E_r^{(t)} = X - \sum_{j \neq i+1, \dots, i+r} \mathbf{d}_j^{(t-1)} \alpha_{rowj}^{(t)}$, 而 $D_r^{(t-1)} A_{rowr}^{(t)} = \mathbf{d}_{i+1}^{(t-1)} \alpha_{rowi}^{(t)} + \dots + \mathbf{d}_{i+r}^{(t-1)} \alpha_{rowi+r}^{(t)}$.

$D_r^{(t-1)}$ 为第 $t-1$ 次迭代字典 $D^{(t-1)}$ 的第 $i+1$ 到 $i+r$ 列 (或者说原子), $A_{rowr}^{(t)}$ 为第 t 次迭代稀疏矩阵 $A^{(t)}$ 的第 $i+1$ 到 $i+r$ 行, $E_r^{(t)}$ 为除了第 $i+1$ 到 $i+r$ 个待求解原子外的残差项. 式中固定 $K-r$ 项 $\sum_{j \neq i, \dots, i+r} \mathbf{d}_j^{(t-1)} \alpha_{rowj}^{(t)}$, 剩下的 r 项 $\mathbf{d}_j^{(t-1)} \alpha_{rowj}^{(t)} + \dots + \mathbf{d}_{j+r}^{(t-1)} \alpha_{rowj+r}^{(t)}$ 是待求解的 ($\mathbf{d}_j^{(t-1)}$ 为第 $t-1$ 次迭代得到字典的第 j 列, $\alpha_{rowj}^{(t)}$ 为第 t 次迭代的稀疏矩阵第 j 行). 同样的, 在字典更新之前需要对信号 X 、残差项 $E_r^{(t)}$ 和稀疏矩阵

$A_{rowr}^{(t)}$ 进行去零收缩, 其方法可参见 K-SVD^[7] 算法的去零收缩, 与以往不同的是本文改进算法是索引 r 行稀疏系数中非零元素所在列, 继而得到 $X^* = X\Omega$ 、 $E_r^{(t)*} = E_r^{(t)}\Omega$ 、 $A_{rowr}^{(t)*} = A_{rowr}^{(t)}\Omega$ 减少了矩阵运算计算量。

与解 MOD 算法模型中相同的方法即最小二乘法, 并结合式 (8) 得到代价函数的解为:

$$D_r^{(t)} = \arg \min_{D_r} \left\| E_r^{(t)*} - D_r^{(t-1)} A_{rowr}^{(t)*} \right\|_F^2 = E_r^{(t)*} A_{rowr}^{(t)*T} (A_{rowr}^{(t)*} A_{rowr}^{(t)*T})^{-1} \quad (9)$$

其中, $A_{rowr}^{(t)*} A_{rowr}^{(t)*T}$ 为 $r \times r$ 的方阵, $A_{rowr}^{(t)*}$ 必须满足行满秩, 否则 $A_{rowr}^{(t)*} A_{rowr}^{(t)*T}$ 不可逆, r 为一次性更新的原子个数. 其证明如下:

证明. 因为矩阵的 F-范数为矩阵全部元素平方和的平方根, 所以对于残差 $\left\| E_r^{(t)*} - D_r^{(t-1)} A_{rowr}^{(t)*} \right\|_F^2$ 可以获得如下变形:

$$\begin{aligned} \left\| E_r^{(t)*} - D_r^{(t-1)} A_{rowr}^{(t)*} \right\|_F^2 &= (E_r^{(t)*} - D_r^{(t-1)} A_{rowr}^{(t)*})^T (E_r^{(t)*} - D_r^{(t-1)} A_{rowr}^{(t)*}) \\ &= (E_r^{(t)*T} - A_{rowr}^{(t)*T} D_r^{(t-1)T}) (E_r^{(t)*} - D_r^{(t-1)} A_{rowr}^{(t)*}) \\ &= E_r^{(t)*T} E_r^{(t)*} - E_r^{(t)*T} D_r^{(t-1)} A_{rowr}^{(t)*} - A_{rowr}^{(t)*T} D_r^{(t-1)T} E_r^{(t)*} + (A_{rowr}^{(t)*T} D_r^{(t-1)T}) (D_r^{(t-1)} A_{rowr}^{(t)*}) \end{aligned} \quad (10)$$

求 $D_r^{(t)} = \arg \min_{D_r} \left\| E_r^{(t)*} - D_r^{(t-1)} A_{rowr}^{(t)*} \right\|_F^2$, 即求残差的极小值问题, 其等价于求残差对变量 $D_r^{(t-1)}$ 导数为 0 的情况. 式 (10) 中, $E_r^{(t)*T} E_r^{(t)*}$ 因为不含 $D_r^{(t-1)}$ 求导为零, $(E_r^{(t)*T}) D_r^{(t-1)} A_{rowr}^{(t)*}$ 和 $(A_{rowr}^{(t)*T} (D_r^{(t-1)T})^T E_r^{(t)*})$ 对 $D_r^{(t-1)}$ 求导都为 $E_r^{(t)*} A_{rowr}^{(t)*}$, $[(A_{rowr}^{(t)*})^T (D_r^{(t-1)T})^T] (D_r^{(t-1)} A_{rowr}^{(t)*})$ 对 $D_r^{(t-1)}$ 求导为 $2D_r^{(t-1)} A_{rowr}^{(t)*} A_{rowr}^{(t)*T}$, 因此将式 (10) 进行求导得:

$$\begin{aligned} \frac{\partial \left\| E_r^{(t)*} - D_r^{(t-1)} A_{rowr}^{(t)*} \right\|_F^2}{\partial D_r^{(t-1)}} &= -E_r^{(t)*} A_{rowr}^{(t)*} - E_r^{(t)*} A_{rowr}^{(t)*} + 2D_r^{(t-1)} A_{rowr}^{(t)*} (A_{rowr}^{(t)*})^T = 0 \end{aligned} \quad (11)$$

将等式 (11) 等价变形为:

$$\begin{aligned} 2E_r^{(t)*} A_{rowr}^{(t)*} &= 2D_r^{(t)} A_{rowr}^{(t)*} (A_{rowr}^{(t)*})^T \Leftrightarrow E_r^{(t)*} A_{rowr}^{(t)*} = D_r^{(t)} A_{rowr}^{(t)*} (A_{rowr}^{(t)*})^T \Leftrightarrow D_r^{(t)} = E_r^{(t)*} A_{rowr}^{(t)*} [A_{rowr}^{(t)*} (A_{rowr}^{(t)*})^T]^{-1} \end{aligned} \quad (12)$$

即证式 (9). \square

用 $D_r^{(t)}$ 代替 $D^{(t-1)}$ 中的 $i+1$ 到 $i+r$ 列, 依次重复所有 $D^{(t-1)}$ 中的原子, 完成一次字典更新. 同 K-SVD 和 SGK 算法一样, 本文算法也是通过稀疏编码 (对 A) 和字典更新 (对 D) 交替重复迭代, 直致达到预定迭代次数或误差要求才结束. 本文算法不仅不会破坏稀疏系数的结构, 而且在顺序更新字典原子的同时大大提高字典学习速度, 多原子同时更新, 减少了单字典原子更新时重复多次迭代的计算时间复杂度. 算法流程图如图 1.

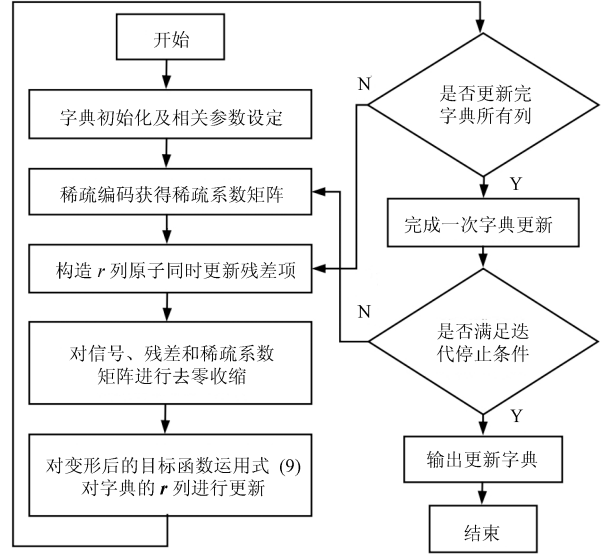


图 1 本字典学习算法流程图

Fig. 1 The flow chart of the dictionary learning algorithm in this paper

改进算法具体运算步骤如下:

- 1) 对迭代停止条件、同时更新的字典列数等参数设定. 字典的初始化, 选择 DCT 字典为初始字典;
- 2) 稀疏编码阶段利用 OMP 算法求得稀疏系数矩阵;
- 3) 利用本文算法进行字典更新;
- 4) 重复步骤 2)、步骤 3), 直到满足迭代停止条件;
- 5) 完成字典训练, 获得字典 D .

3 实验结果分析

本文的所有数据实验均在 MATLAB R2014a 进行编程运行, 试验用计算机配置为: 处理器为主频 3.3 GHz 的 AMD FX8300、8 GB 内存, 操作系统为 Windows 7 64 位. 为验证压缩测量后保存图像信息的完整性, 本文将测量后的信号进行重构, 重构图像质量的好坏代表了获得字典质量的优劣. 原始图像大小为 256×256 , 字典大小为 64×256 , 当显示每个原子字典为图像时, 每个原子占据一个 8×8 像

素的图像单元. OMP 算法中重构信号允许表示误差为 $\sigma \times C$, $C = 1.15$, σ 为加入高斯白噪声的标准差. 稀疏字典是从噪声图像块自身训练出来的, 字典学习过程包含了对每块大小为 8×8 的噪声图像块的稀疏编码步骤. 为了方便对图像重构效果直观地比较, 一般情况下取字典训练迭代次数为 10 次, 取学习字典的初始字典为 DCT 字典, 图像重构算法都采用 OMP 算法. 实验内容主要包括: 1) 本文算法性能及其参数分析, 包括待更新字典原子的选取方式和同时更新字典的列数; 2) K-SVD、SGK 和本文算法结果对比验证, 包括不同噪声条件和不同迭代次数下各算法效果对比; 3) 对不同来源图像情况下算法效果的稳定性比较.

3.1 重构图像的评价

本文采用峰值信噪比 PSNR (Peak signal to noise ratio) 来衡量图像重构质量, 其定义为:

$$MSE = \frac{1}{mn} \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} \left\| \hat{X}(i, j) - X(i, j) \right\|^2 \quad (13)$$

$$PSNR = 20 \cdot \lg \left(\frac{X_{\max}}{\sqrt{MSE}} \right) \quad (14)$$

其中, $\hat{X}(i, j)$ 为重构图像, $X(i, j)$ 为原始图像, X_{\max} 是表示图像点颜色的最大数值, 原图每个采样点用 8 位表示, 则 X_{\max} 取 255, PSNR 的单位为 dB. PSNR 值越大, 就代表失真越少, 重构效果就越好. PSNR 是图像空间域统计特性评价方法, 可以提供和主观感知较为相统一的判断, 是一种性能良好的噪声失真评价标准; 而对于结构性失真, 由于其并未考虑人眼的视觉特性, 不能有效评判图像质量.

结构相似度 SSIM^[14] (Structural similarity Index) 是一种衡量两幅图像相似性的指标. SSIM 计算了亮度、对比度和结构三部分的失真, 通过获得分块图像的失真度量, 最后求均值获得整幅图像的失真度. 图像块结构相似度 SSIM 和平均结构相似度 MSSIM 定义如下:

$$SSIM(\mathbf{x}, \hat{\mathbf{x}}) = \frac{(2\mu_{\hat{\mathbf{x}}}\mu_{\mathbf{x}} + C_1)(2\delta_{\mathbf{x}\hat{\mathbf{x}}} + C_2)}{(\mu_{\hat{\mathbf{x}}}^2 + \mu_{\mathbf{x}}^2 + C_1)(\delta_{\hat{\mathbf{x}}}^2 + \delta_{\mathbf{x}}^2 + C_2)} \quad (15)$$

$$MSSIM(X, \hat{X}) = \frac{1}{M} \sum_{m=1}^M SSIM(\mathbf{x}_m, \hat{\mathbf{x}}_m) \quad (16)$$

$\mu_{\mathbf{x}}$ 、 $\mu_{\hat{\mathbf{x}}}$ 分别表示图像块 \mathbf{x} 和 $\hat{\mathbf{x}}$ 的均值, $\delta_{\mathbf{x}}$ 、 $\delta_{\hat{\mathbf{x}}}$ 分别表示图像块 \mathbf{x} 和 $\hat{\mathbf{x}}$ 的方差, $\delta_{\mathbf{x}\hat{\mathbf{x}}}$ 表示其协方差, C_1 、 C_2 为常数, M 为一幅图像分块总数. MSSIM 取值范围 $[0, 1]$, 值越大, 表示图像失真越小, 当两幅图像完全一样时值为 1.

3.2 本文算法性能及其参数分析

文首先选取 house 图像为仿真实验用图像. 图 2 为原始图像和加入噪声图像对比图, 高斯白噪声标准差 σ 为 25, 加噪后图像 PSNR 值为 20.1875 dB. 本文算法在字典更新过程中, 采用字典的 r 列原子同时更新, 但选取同时更新原子的次序对字典更新时间和重构图像效果有很大影响. 用于更新的字典原子选取的方式有顺次选取连续的 r 列、随机选取连续的 r 列和随机选取离散的 r 列这 3 种方式. 为方便比较以上三种字典原子选取方式暂时选取 $r = 2$ 列原子同时更新的情况.

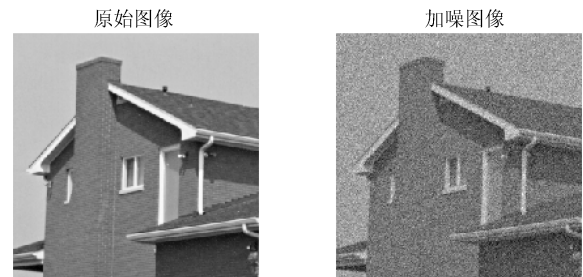


图 2 原始图像及其加噪图像

Fig. 2 Original image and noisy image

由表 1 可以看出选取待更新字典原子时, 用顺次选取连续 r 列方法的字典训练时间要明显短于随机选取连续 r 列和随机选取离散 r 列, 而且从重构图像质量上, 顺次选取连续 r 列方法的峰值信噪比和平均结构相似度也明显高于其他两种方式. 因此, 本文算法在下面的实验中采用的字典原子选择方式都为顺序选取连续的 r 列的方式.

为了不重复且完整地更新完字典所有的列, 同时更新字典原子列数必须可以被字典总列数整除, 由于字典原子个数为 256, 所以一次更新原子数量可以为 2、4、8、16 等. 当同时更新原子列数较少时, 虽然更新次数较多, 但在式 (9) 中求逆矩阵的计算复杂度较小; 在同时更新原子列数很大时, 求逆矩阵的计算复杂度变大, 但也同时减少了多次迭代的计算量. 如表 2 所示, 在同时更新字典原子个数不同的情况下, 从字典学习速度和重构图像质量考虑, 在保证重构图像质量基本不变的情况下同时更新 8 列或 16 列原子其字典学习时间具有明显优势. 为了简单直观显示本文算法优势, 在下面的实验中将字典同时更新的原子数定为 8 列.

3.3 算法对比分析

本文采用 DCT 字典、K-SVD 字典、SGK 字典和本文方法进行仿真对比, 以 house 图像为例, 在加入标准差 σ 为 25 的高斯白噪声、字典迭代次数为 10 次、其他条件相同的情况下, 得到的重构效果对比如图 3. 由图 3 和与之对应的表 3 可知用 K-SVD

字典、SGK 字典与本文算法学习出的字典进行稀疏表示图像重构, 其图像质量相近, 且明显好于离散余弦字典.

在字典学习时间上, 分别通过 10 次完整的运算求运算时间平均值. SGK 字典学习算法是 K-SVD 算法所用时间的 21%, 而本文算法是 K-SVD 算法所用时间的 18%, 相当于为 SGK 字典学习算法的 85%, 进一步提高了字典学习效率, 降低了字典学习的计算复杂度.

在稀疏表示的过程中, 字典训练的效果随迭代次数的增加而不断提高, 但训练所消耗的时间也随之不断变长, 为探究训练随迭代次数增加字典更新的效果进行如下实验. 图 4 和表 4 实验以标准差 σ 为 25 的高斯白噪声 house 图像为实验对象, 字典迭代次数为 3~21 次. 本文算法相对于 K-SVD 算法、SGK 算法, 其运算速度有较明显的提高, 在迭代次数增加的情况下其字典训练时间差距不断变大如图 4 所示.

表 1 不同字典原子选取方式效果对比

Table 1 Comparison of the effects for different selections of dictionary atoms

选取待更新字典原子的方式	顺次选取连续的 2 列	随机选取连续的 2 列	随机选取离散的 2 列
字典学习时间 (s)	3.994	5.268	6.788
峰值信噪比 PSNR (dB)	32.092	31.968	31.866
平均结构相似度 MSSIM	0.3183	0.3152	0.3114

表 2 同时更新不同字典列数效果对比

Table 2 Effect comparison of updating different numbers of dictionary columns at the same time

同时更新原子列数	2	4	8	16	32	64
字典学习时间 (s)	4.117	4.194	3.630	3.744	4.032	4.093
峰值信噪比 PSNR (dB)	32.092	32.065	32.081	32.145	32.058	32.045
平均结构相似度 MSSIM	0.3183	0.3176	0.3177	0.3173	0.3164	0.3181

表 3 不同算法的重构效果

Table 3 Reconstruction effect with different algorithms

算法	DCT 字典	K-SVD 字典	SGK 字典	本文算法字典
字典学习时间 (s)	-	20.079	4.254	3.605
峰值信噪比 PSNR (dB)	30.915	31.950	31.935	31.947
平均结构相似度 MSSIM	0.3022	0.3149	0.3137	0.3144

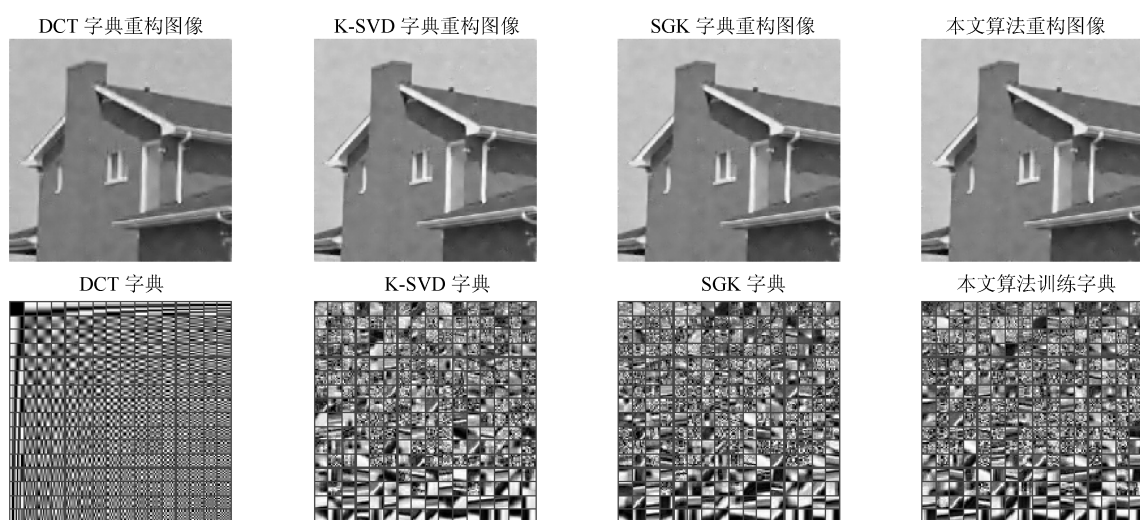


图 3 重构效果对比图

Fig. 3 The reconstruction effect comparison

表4 不同迭代次数重构图像质量

Table 4 The quality of the reconstructed image with different numbers of iteration

迭代次数	K-SVD 字典		SGK 字典		本文算法字典	
	PSNR (dB)	MSSIM	PSNR (dB)	MSSIM	PSNR (dB)	MSSIM
3	31.4559	0.30762	31.4459	0.30779	31.4603	0.30825
6	31.9951	0.31384	31.9291	0.31171	31.9265	0.31207
9	31.9704	0.31386	31.9490	0.31287	31.9194	0.31313
12	32.2901	0.31418	32.2661	0.31423	32.2416	0.31279
15	32.3571	0.31986	32.3297	0.31898	32.2724	0.31921
18	32.4407	0.32208	32.3781	0.32182	32.3485	0.32141
21	32.4489	0.32209	32.4040	0.32206	32.3971	0.32172

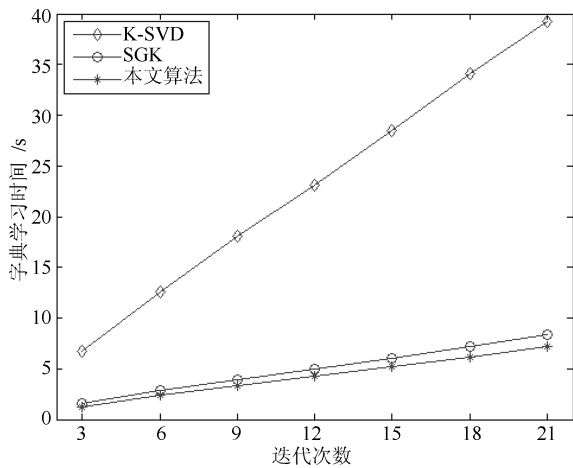


图4 不同迭代次数下的字典学习时间

Fig. 4 Time consumption of dictionary learning for different iteration times

对于在字典学习基础上的图像压缩重构算法,字典更新迭代次数的增加使图像质量也随之提高.由表4可以看出,本文算法相比于K-SVD算法、SGK算法,其图像重构效果十分接近.结合图4,在迭代次数不同的情况下可以得出本文算法相比于传统方法,其重构图像质量相同且字典学习效率有了较明显提高.

在现实场景中,噪声的影响在机器人图像中是十分重要的因素,用通过训练得到的字典来重构图像具有一定去噪效果.图5和表5实验以加入标准差 $10 \leq \sigma \leq 70$ 高斯白噪声的 house 图像为研究对象,在其他条件与之前相同情况下,通过对不同程度噪声下各个字典学习算法的训练时间和图像重构效果进行比较验证本文算法的优势.

图5可知,随着加入噪声的标准差的变大,字典训练时间在不断变短.当噪声标准差大于30后,本文字典训练时间开始明显低于SGK算法.在 $\sigma = 50$ 时达到基本稳定,此时本文算法完成10次迭代更新字典耗时2.679s, K-SVD算法和SGK算法分别耗时17.996s、3.607s,本文算法仅为K-SVD

算法耗时的14.8%和SGK算法的74.3%,降低了计算复杂度,节省了字典更新的时间.

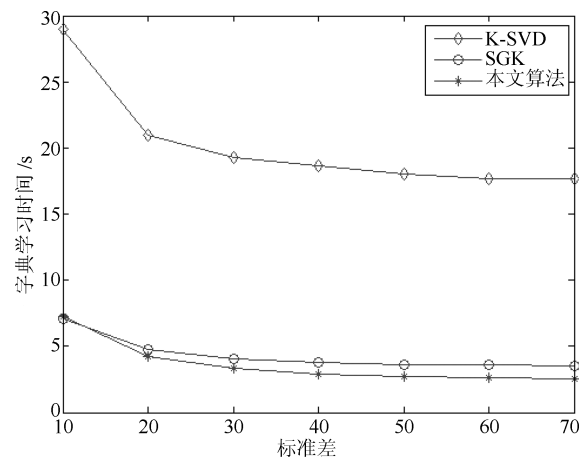


图5 不同噪声程度下的字典学习时间

Fig. 5 Time consumption of dictionary learning for different noise levels

从表5可以看出,随着加入噪声的标准差的变大,重构图像的质量也在变差,但随噪声的增大,本文算法重构图像质量略微优于其他两种方法,也就是说仅从实验数据上看在较强噪声环境下,用本文算法获得的字典重构图像的质量能够达到甚至优于K-SVD算法和SGK算法.

3.4 不同图像下算法效果比较

为进一步验证本文改进算法对图像信号的稀疏分解效果,将常用于图像处理的包含不同纹理的 lena、peppers、barbara 图像用于K-SVD字典、SGK字典与本文算法的字典训练和稀疏分解中,实验中统一将其处理为 256×256 的图像,本文算法中标准差取 $\sigma = 50$,顺次连续选取同时更新的字典原子数为8列,各个算法迭代次数都为10次,其他参数保持不变,比较其重构图像质量和完成字典学习所用时间如表6所示.

表 5 不同噪声水平下重构图像质量

Table 5 The quality of the reconstructed image with different noise levels

标准差	K-SVD 字典		SGK 字典		本文算法字典	
	PSNR (dB)	MSSIM	PSNR (dB)	MSSIM	PSNR (dB)	MSSIM
10	35.9741	0.45381	35.9440	0.45390	35.9620	0.45430
20	33.2254	0.34207	33.1664	0.34219	33.2006	0.34282
30	31.3615	0.30139	31.2992	0.29996	31.3157	0.29984
40	29.4226	0.26534	29.3889	0.26387	29.4074	0.26427
50	28.0932	0.23598	28.0699	0.23470	28.0908	0.23485
60	26.8134	0.20862	26.8110	0.20815	26.8230	0.20744
70	25.6137	0.18573	25.6164	0.18493	25.6238	0.18591

表 6 不同图像信号的重构效果

Table 6 Reconstruction effect of different image signals

图像	算法	时间 (s)	MSSIM	PSNR (dB)
lena	K-SVD	18.343162	0.27010	25.8505
	SGK	3.891636	0.26863	25.8154
	本文算法	3.102337	0.26916	25.8230
peppers	K-SVD	19.244135	0.35640	26.1407
	SGK	4.141303	0.35574	26.1445
	本文算法	3.092842	0.35545	26.1353
barbara	K-SVD	19.438234	0.35642	25.2067
	SGK	4.114145	0.35763	25.1997
	本文算法	3.197942	0.35774	25.2009

表 6 中数据为通过 10 次实验求得平均值, 从表中可以看出, 本文算法能够有效降低字典训练的时间, 且图像的重构质量相比其他两种算法并无明显差距. 在不同纹理复杂度图片情况下, 本文算法相比 SGK 算法, lena 图像的字典训练时间减少了 20.3%, 平均结构相似度提高了 0.00053, 图像峰值信噪比提高了 0.0076 dB; peppers 图像字典训练时间减少了 25.3%, 平均结构相似度下降了 0.00029, 图像峰值信噪比下降了 0.0092 dB; barbara 图像字典训练时间下降了 22.3%, 平均结构相似度提高了 0.00011, 图像峰值信噪比提高了 0.00127 dB. 仿真结果表明本文算法能够在图像质量变化很小的前提下, 有效缩短字典学习所耗费的时间, 降低字典学习过程的计算复杂度, 提高图像压缩感知的效率.

由于机器人通常所在的环境和所处理的图像为自然场景或自然图像, 其复杂度高, 场景随机性大, 所以本文最后将处理的对象变成在不同环境中获得的场景图像, 并提前将其处理成尺寸为 256×256 的灰度图像应用于仿真实验. 如图 6 所示经过 OMP 算法重构获得三个对比场景图像重构效果对比图, (a) 为明暗场景, (b) 为不同视角场景图, (c) 为模

糊清晰场景图. 对上述三种场景图像分别应用 K-SVD、SGK 和本文算法进行字典训练, 试验参数同 lena 等 3 张图片相同.

图 6(a) 和表 7 中比较了在明暗两种情况下本文算法相对于其他两种算法的效果, 实验结果表明在图像变暗的情况下字典训练速度有所加快, 单从字典学习方法来说, 在图像质量变化较小的情况下 (MSSIM 差距不超过 0.001, PSNR 差距不超过 0.02 dB) 本文算法无论在何种环境中本文字典学习速度都优于其他两种算法. 由图 6(b) 和表 7 可以看出视角 1 字典训练速度比视角 2 快, 但在同一视角来说 MSSIM 差距不超过 0.001、PSNR 不超过 0.02 dB, 本文算法字典训练耗时是 K-SVD 算法的 15%~20%、是 SGK 算法的 75%~85%. 图 6(c) 和表 7 表明在图像越不清楚的情况下字典学习速度越快, 其同一清晰度下本文算法与另两种算法重构图像的 MSSIM 差距不超过 0.005、PSNR 不超过 0.01 dB, 字典学习速度明显提高, 耗时显著减少.

4 计算复杂度分析

本文仅对字典更新阶段进行算法的复杂度分析, 比较了本文算法和 SGK 算法字典更新阶段所需要浮点运算次数. 为了计算时间复杂度本文假设每个长度为 n 的信号可以用 m 个非零项稀疏表示, 且信号 X 中有 N 个这样的训练信号. 为了更公平地比较, 我们讨论就参数 K 而言的所有变化.

一般来说信号的维数 $n = O(K)$, 训练样本的数量 $N = O(K^{1+a})$, 其中 $a \geq 0$. 对于获得最小复杂度条件可以通过对稀疏度 $m = O(K^b)$ 推算得到, 稀疏系数向量中非零原子所在列的标号 j 提取出来, 定义标号数组 $\omega_k = \{j | \alpha_{row_j}^{(t)}(j) \neq 0, 1 \leq j \leq N\}$, 再定义矩阵 ω , 并令其元素 $(\omega_k(j), j)$ 为 1, 其他位置元素为 0. 更新完所有原子一次所需非零原子个数为 $\sum_K |\omega| = mN$.

SGK 算法是单列字典原子依次更新的, 其中 $E_k^{(t)} = X - \sum_{j \neq k} d_j^{(t)} \alpha_{row_j}^{(t)}$ 的求解需要 $2nm |\omega|$

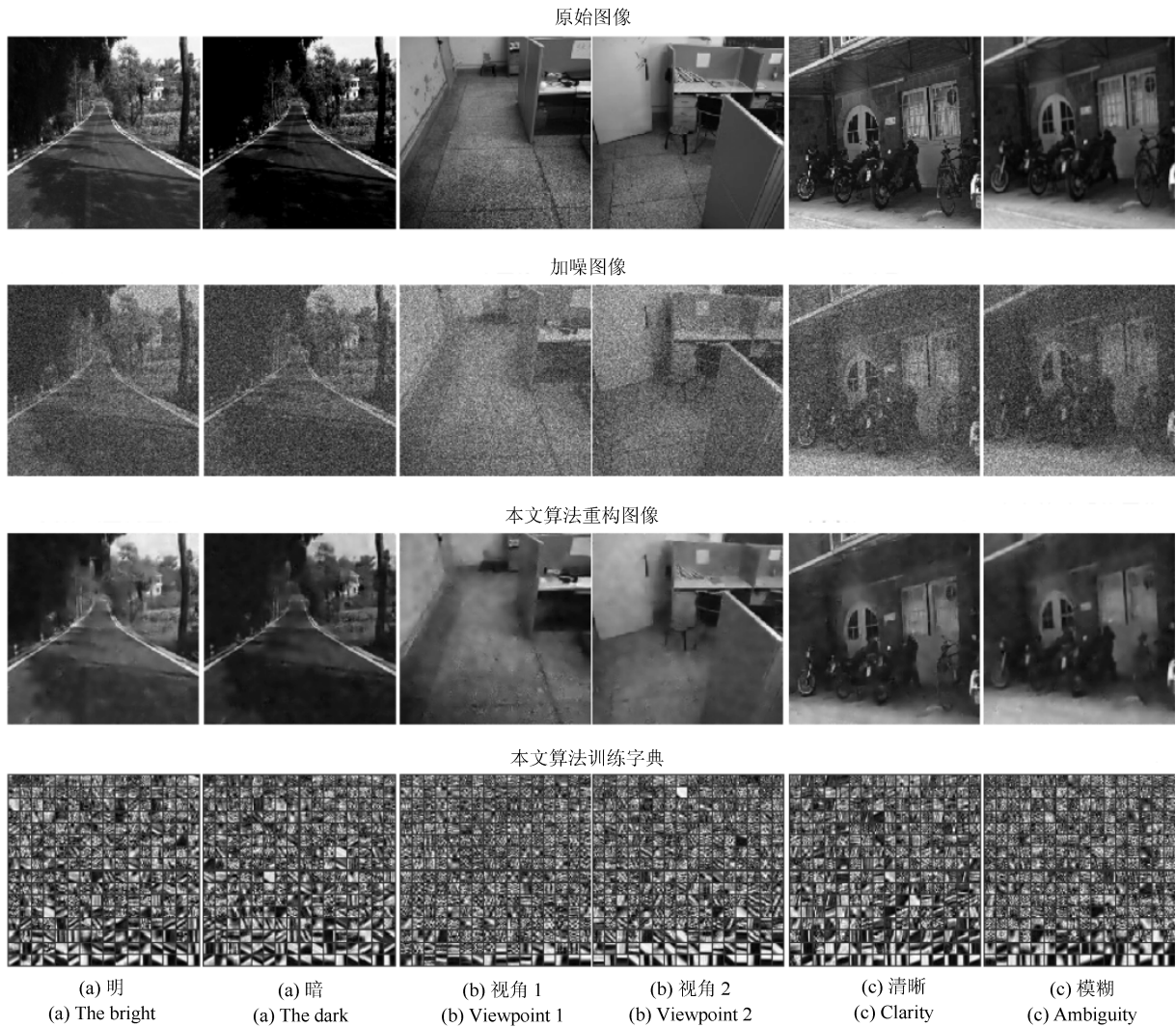


图 6 自然场景图像重构效果对比

Fig. 6 Comparison of image reconstruction effects in natural scenes

表 7 不同场景自然图像信号的重构效果

Table 7 Reconstruction effect of image signals in different natural scenes

图像	时间 (s)			MSSIM			PSNR (dB)		
	K-SVD	SGK	本文算法	K-SVD	SGK	本文算法	K-SVD	SGK	本文算法
(a) 明	20.196063	4.470679	3.958860	0.35006	0.35036	0.35019	26.3382	26.3285	26.3339
(a) 暗	19.138866	4.303163	3.779787	0.25134	0.25155	0.25157	27.0640	27.0637	27.0611
(b) 视角 1	19.294378	4.068634	3.263159	0.29148	0.29095	0.29066	25.9845	25.9772	29.9781
(b) 视角 2	21.498433	4.167165	3.233882	0.30110	0.30121	0.30176	28.2839	28.2803	28.2971
(c) 清晰	20.856516	4.534554	4.031721	0.35926	0.35839	0.35854	26.0795	26.0734	26.0601
(c) 模糊	18.319955	3.874162	2.932584	0.41004	0.41013	0.40993	29.4348	29.4386	29.4385

次浮点运算, $E_k^{(t)}(\alpha_{rowk}^{(t)})^T$ 需要 $n(2|\omega| - 1)$ 次运算, $\alpha_{rowk}^{(t)}(\alpha_{rowk}^{(t)})^T$ 需要 $2|\omega| - 1$ 次运算, 求得的 $\alpha_{rowk}^{(t)}(\alpha_{rowk}^{(t)})^T$ 为一个非负数值, 求其逆即求倒数运算 $[\alpha_{rowk}^{(t)}(\alpha_{rowk}^{(t)})^T]^{-1}$ 需要进行 n 次浮点运算. 综上, 完成一系列字典原子更新需要进行 $2nm|\Omega| + 2|\Omega| - 1$ 次浮点运算, 那么完整更新一次字典的所有列需要

的浮点运算次数为

$$T_{SGK} = \sum_K 2nm|\Omega| + 2|\Omega| - 1 = 2nm^2N + 2mN - K \quad (17)$$

对于本文所提出的算法需要对多列字典原子

同时更新. 假设同时更新 r 列字典原子, $E_r^{(t)} = X - \sum_{j \neq k \dots k+r} \mathbf{d}_j^{(t)} \boldsymbol{\alpha}_{rowj}^{(t)}$ 需要 $n|\Omega|(m-r)$ 次运算求和与 $n|\Omega|(m-r)$ 次运算乘积, 总共也就是 $2n|\Omega|(m-r)$ 次浮点运算; 对于 $E_r^{(t)}(A_{rowr}^{(t)})^T$ 需要 $nr(|\Omega|-1)$ 次运算求和与 $nr|\Omega|$ 次运算乘积, 总共也就是 $2nr|\Omega| - nr$ 次浮点运算; $A_{rowr}^{(t)}(A_{rowr}^{(t)})^T$ 需要 $r^2(|\Omega|-1)$ 次运算求和与 $r^2|\Omega|$ 次运算乘积, 总共也就是 $2r^2|\Omega| - r^2$ 次浮点运算; 得到的 $A_{rowr}^{(t)}(A_{rowr}^{(t)})^T$ 是一个 $r \times r$ 的正定矩阵, 求 $[A_{rowr}^{(t)}(A_{rowr}^{(t)})^T]^{-1}$ 需要 $r^3/2 - r^2/2$ 次求和、 $r^3/2 + 3r^2/2$ 次求积和 r 次运算来进行平方根计算, 即 $r^3 + r^2 + r$ 次浮点运算. 综上合计, 更新 r 列字典原子需要 $(2mn + 2r^2)|\Omega| - nr + r^3 + r$ 次浮点运算, 则更新所有的字典列需要浮点运算次数为

$$T_r = \sum_{\frac{K}{r}} (2mn + 2r^2)|\Omega| - nr + r^3 + r = \frac{2nm^2N}{r} + 2mNr + K(1 + r^2 - n) \quad (18)$$

为更清楚比较本文算法和原始 SGK 算法在字典更新过程中的计算复杂度, 需要获得字典 $D \in \mathbf{R}^{n \times K}$ 的具体尺寸 $n = 64$ 、 $K = 256$ 和稀疏系数行向量 $\boldsymbol{\alpha}_{rowk}^{(t)} \in \mathbf{R}^N$ 维数或训练信号集 $X = \{\mathbf{x}_1, \dots, \mathbf{x}_N\} \in \mathbf{R}^{n \times N}$ 中信号的个数 $N = 62001$, 稀疏度水平 m 的选择本文取 $m = 1$. 若本文算法能减小计算复杂度则 $T_\Delta = T_r - T_{SGK} < 0$.

$$T_\Delta = 2nm^2N \left(\frac{1}{r} - 1 \right) + 2mN(r-1) + K(2 + r^2 - n) = \frac{7936128}{r} + 124002r + 256r^2 - 8044258 \quad (19)$$

其中, $r = 2, 4, 8, 16, \dots, 256$. 对 T_Δ 求最小值得 $T_{\Delta \min} = -6.0441 \times 10^6$, $r_{\min} = 7.873$. 因此, 同时更新字典列数 $r = 8$ 列时计算复杂度最低, 约减少 600 万次浮点运算. 选取不同参数情况下结果有所不同, 但 $T_\Delta < 0$ 证明了通过本文算法确实能降低字典更新阶段计算复杂度.

5 结论

对机器人图像压缩采样过程中数据计算量大、计算时间长的问题, 引入压缩感知算法, 改进了原始学习字典稀疏表示计算时间复杂度大、耗时长的问题. 改进的算法是对 SGK 算法效率的进一步提高, 从参与更新的字典原子列数入手, 将残差项变形为多列原子同时更新, 进而利用最小二乘法连续地更新字典中的多个原子, 减少了单次迭代的计算量, 提

高了字典学习效率. 通过学习获得的冗余字典具有更好的重建图像的能力, 字典中原子与待重建图像具有相关性, 能更有效表达图像局部结构特点, 使得信号在字典上有更稀疏的表示. 利用不同情况下获得的图像, 在相同计算参数下, 本文算法相比于原 SGK 字典学习算法, 压缩采样后保存了基本相同的信息量, 基于各自冗余字典重构的图像质量近似相同, 且字典学习计算速度提高明显, 相比于主流的 K-SVD 算法更大大提高了字典的学习效率.

References

- 1 Candes E J. Compressive sampling. In: Proceedings of International Congress of Mathematicians. Madrid, Spain: European Mathematical Society Publishing House, 2006. 1433–1452
- 2 Ren Yue-Mei, Zhang Yan-Ning, Li Ying. Advances and perspective on compressed sensing and application on image processing. *Acta Automatica Sinica*, 2014, **40**(8): 1563–1575
(任越美, 张艳宁, 李映. 压缩感知及其图像处理应用研究进展与展望. *自动化学报*, 2014, **40**(8): 1563–1575)
- 3 Zhan Xin. Research on Dictionary Learning-based SAR Image Compression Techbique [Ph. D. dissertation], University of Science and Technology of China, China, 2015
(占新. 基于字典学习与稀疏模型的 SAR 图像压缩技术研究 [Ph. D. dissertation]. 中国科学技术大学, 2015)
- 4 Olshausen B A, Field D J. Emergence of simple-cell receptive field properties by learning a sparse code for natural images. *Nature*, 1996, **381**(6583): 607–609
- 5 Engan K, Aase S O, Husoy J H. Method of optimal directions for frame design. In: Proceedings of the 1999 IEEE International Conference on Acoustics, Speech, and Signal Processing. Phoenix, AZ, USA: IEEE, 1999. 2443–2446
- 6 Aharon M, Elad M, Bruckstein A M. The K-SVD: an algorithm for designing of overcomplete dictionaries for sparse representation. *IEEE Transactions on Signal Processing*, 2006, **54**(11): 4311–4322
- 7 Wang Qiang, Zhang Pei-Lin, Wang Huai-Guang, Wu Ding-Hai, Zhang Yun-Qiang. Data compression algorithm of vibration signal based on sparse decomposition. *Chinese Journal of Scientific Instrument*, 2016, **37**(11): 2497–2505
(王强, 张培林, 王怀光, 吴定海, 张云强. 基于稀疏分解的振动信号数据压缩算法. *仪器仪表学报*, 2016, **37**(11): 2497–2505)
- 8 Yi Ke-Fu, Wang Dong-Hao, Wan Jiang-Wen. Optimized dictionary learning algorithm for compressive data gathering. *Journal of Beijing University of Aeronautics and Astronautics*, 2016, **42**(6): 1203–1209
(易可夫, 王东豪, 万江文. 基于优化字典学习算法的压缩数据收集. *北京航空航天大学学报*, 2016, **42**(6): 1203–1209)
- 9 Lu Wan-Yun, Wang Ji-Zhou, Cao-Meng. Cloud removal in ZY-3 remote sensing image based on atoms-reordered dictionary learning AO-DL. *Acta Geodaetica et Cartographica Sinica*, 2017, **46**(5): 623–630
(陆婉芸, 王继周, 曹萌. 变序字典学习 AO-DL 的资源三号遥感影像云去除. *测绘学报*, 2017, **46**(5): 623–630)

- 10 Sahoo S K, Makur A. Replacing K-SVD with SGK: Dictionary training for sparse representation of images. In: Proceedings of the 2015 IEEE International Conference on Digital Signal Processing. Singapore, Singapore: IEEE, 2015. 614–617
- 11 Lian Qiu-Sheng, Shi Bao-Shun, Chen Shu-Zhen. Research advances on dictionary learning models, algorithms and applications. *Acta Automatica Sinica*, 2015, **41**(2): 240–260 (练秋生, 石保顺, 陈书贞. 字典学习模型、算法及其应用研究进展. *自动化学报*, 2015, **41**(2): 240–260)
- 12 Sahoo S K, Makur A. Dictionary training for sparse representation as generalization of K-means clustering. *IEEE Signal Processing Letters*, 2013, **20**(6): 587–590
- 13 Sahoo S K, Makur A. Image denoising via sparse representations over Sequential Generalization of K-means (SGK). In: Proceedings of the 9th IEEE International Conference on Information, Communications and Signal Processing. Tainan, China: IEEE, 2014. 1–5
- 14 Wang Z, Bovik A, Sheikh H, et al. Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing*, 2004, **13**(4): 600–612

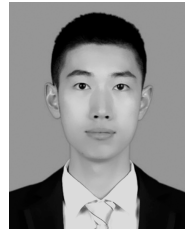


郭俊锋 兰州理工大学机电工程学院副教授. 主要研究方向为先进控制技术, 信号检测, 压缩感知.

E-mail: junf_guo@163.com

(**GUO Jun-Feng** Associate professor at the College of mechanical and electrical engineering, Lanzhou University of Technology. His research interest

covers advanced control technology, signal detection, and compression sensing.)



李育亮 兰州理工大学硕士研究生. 主要研究方向为机器人图像处理, 稀疏表示算法. 本文通信作者.

E-mail: ly1931206@163.com

(**LI Yu-Liang** Master student at the College of mechanical and electrical engineering, Lanzhou University of Technology. His research interest covers

robot image processing and sparse representation algorithm. Corresponding author of this paper.)