

基于预学习字典的对称ADMM在人脸去噪上的应用

张静雯, 党亚峥, 倪诗皓

上海理工大学管理学院, 上海

收稿日期: 2024年1月29日; 录用日期: 2024年3月21日; 发布日期: 2024年3月28日

摘要

字典学习算法是图像去噪的常用方法, 但字典训练是一个耗时的过程, 且会丢失图像中的有效信息。本文提出一种基于预学习字典的低秩算法, 将图像中的干净数据和结构化噪声用对应的字典和系数矩阵表示。首先通过消融实验证明模型各部分的作用, 接着对部分损坏的图像进行训练并与其他算法对比, 最后将算法训练出的映射矩阵用于测试数据。结果表明PLID对于噪声的处理优于其他算法, 测试数据经过映射矩阵的线性变换后也可以去除噪声。

关键词

预学习字典, 对称ADMM, 人脸去噪, 消融实验

Application of Symmetric ADMM Based on Pre-Learned Dictionary in Face Denoising

Jingwen Zhang, Yazheng Dang, Shihao Ni

Business School, University of Shanghai for Science and Technology, Shanghai

Received: Jan. 29th, 2024; accepted: Mar. 21st, 2024; published: Mar. 28th, 2024

Abstract

Dictionary learning algorithm is a common method for image denoising, but dictionary training is a time-consuming process and can lose valid information in images. In this paper, a low-rank algorithm based on pre-learned dictionaries is proposed, which represents the clean data and structured noise in the image with the corresponding dictionaries and coefficient matrices. Firstly, the role of each part of the model is proved by ablation study, then the partially damaged images are trained and compared with other algorithms. Finally, the mapping matrix trained by the algorithm

is used for the test data. The results show that PLID can deal with the noise better than other algorithms, and the test data can also remove the noise after the linear transformation of the mapping matrix.

Keywords

Pre-Learned Dictionaries, Symmetric ADMM, Face Denoising, Ablation Study

Copyright © 2024 by author(s) and Hans Publishers Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

1. 引言

随着机器学习和稀疏表示的重大进步，图像处理已成为一个重要的研究课题，由于受到成像设备或外界环境变化等因素的影响，图像中难免会存在不同程度的噪声。人们为了从图像中提取更多的有用信息，运用图像去噪技术来去除噪声。

字典学习是实现图像去噪的一种有效技术，但在传统的字典学习(K-SVD)中，字典训练是一个耗时的过程，稀疏表示大多通过设置固定的硬阈值来停止，导致图像中的有效内容丢失[1]。基于上述问题，一些研究对传统的字典学习算法进行改进，文献[2]提出了一种改进的 K-SVD 算法，用于稀疏表示去噪，该算法具有更快的执行速度。Song 等人[3]提供了一种参数化的模糊自适应字典改编方法，嵌入了模糊集的新机制，将字典列的更新与稀疏表示的更新相结合。然而，基于 K-SVD 的去噪算法仅利用图像块的内部信息进行独立的稀疏编码，而不考虑与其他块相关的信息。

数据不同的组成部分可能具有不同的结构，它们位于不同的子空间中可以单独表示[4]，因此可以根据工作的特性构造特定的字典，这样可以有效地保留原始数据之间的局部几何结构[5]。基于此类想法，Chen 等人[6]将 RPCA 的稀疏项替换为预学习的语音字典和系数矩阵的乘积，将背景噪声建模为低秩部分和残差之和，从而实现语音增强。Xu 等人[7]基于预学习字典和自适应参数设置方法，选择平均重构质量最好的字典作为预学习字典来降低计算的复杂性，实验表明该算法的性能优于其他去噪算法。Wright 等人[8]指出人脸的干净数据可以很好地用子空间来表征，且同一类的干净数据位于同一低维子空间中。此外，结构化噪声通常对应于人脸图像[9]中的面具、眼镜等，可以用字典和对应系数的乘积来表示。

综上所述，本文提出了一种基于预学习字典的低秩算法用于人脸去噪，在算法中，干净数据和结构化噪声分别被建模为它们对应的字典和系数矩阵的乘积，并进一步描述重构误差与干净数据的联系，然后利用对称 ADMM 对所提的算法进行求解，最后，在 AR 数据集上验证所提算法的有效性。

2. 相关理论与方法

2.1. RPCA

RPCA [10]旨在从损坏的观测矩阵 X 中恢复低秩矩阵 L ，其数学模型为：

$$\begin{aligned} \min_{L,S} \text{rank}(L) + \lambda \|S\|_0 \\ \text{s.t. } X = L + S \end{aligned} \quad (2.1)$$

其中， $\text{rank}(L)$ 为矩阵的秩， $\|S\|_0$ 表示矩阵 S 中非零元素的个数。问题(2.1)是一个 NP-难问题，在实际应

用中通常转化为问题(2.2)

$$\begin{aligned} \min_{L,S} & \|L\|_* + \lambda \|S\|_1 \\ \text{s.t.} & X = L + S \end{aligned} \quad (2.2)$$

其中 $\|L\|_*$ 是矩阵 L 的奇异值的总和。 $\|S\|_1$ 是矩阵中每个列向量的绝对值之和的最大值。

2.2. 聚类评价指标

评判聚类结果的好坏,除了凭直观感受评价可视化的结果外,还可以借助数值化的聚类有效性指标,本文使用的评价指标有 ACC, PUR [11]、NMI [12]。

2.2.1. PUR

纯度(Purity)为正确聚类的样本个数占总样本数的比例,可表示为

$$Purity(\Omega, C) = \frac{1}{N} \sum_{i=1}^k \max_j |w_k \cap c_j| \quad (2.3)$$

其中, N 表示总的样本个数, $\Omega = \{w_1, w_2, \dots, w_k\}$ 表示正确的聚类结果, $C = \{c_1, c_2, \dots, c_j\}$ 表示真实的聚类结果。 $PUR \in [0,1]$, 越接近 1 表示聚类的结果越好。

2.2.2. NMI

假定 X 是一个由 n 个样本点组成的样本集 $X = \{x_1, x_2, \dots, x_n\}$, $P = \{C_1, C_2, \dots, C_K\}$ 是正确的聚类结果, $Q = \{C'_1, C'_2, \dots, C'_K\}$ 是待和 P 比较的聚类结果。 P 和 Q 之间 NMI 指标(normalized mutual information)可表示为

$$NMI(P, Q) = \frac{2 \sum_{h=1}^K \sum_{l=1}^{K'} \frac{|C_h \cap C'_l|}{n} \log \frac{|C_h \cap C'_l|}{n |C_h| |C'_l|}}{\sum_{h=1}^K \frac{|C_h|}{n} \log \frac{|C_h|}{n} + \sum_{l=1}^{K'} \frac{|C'_l|}{n} \log \frac{|C'_l|}{n}} \quad (2.4)$$

NMI 指标的最大值为 1, 最小值为 0, 值越大表示聚类结果越接近真实结果。

2.2.3. ACC

准确度(Accuracy)使用正确的聚类结果与真实的聚类结果进行匹配,可表示为

$$AC = \frac{\sum_{i=1}^n \delta(s_i, \text{map}(r_i))}{n} \quad (2.5)$$

其中, r_i , s_i 分别表示 x_i 所对应的正确聚类结果和真实聚类结果, n 表示总样本数, δ 表示指示函数,定义如下:

$$\delta(x, y) = \begin{cases} 1, & x = y \\ 0, & \text{otherwise} \end{cases}$$

ACC 的取值范围为 0 到 1, 值越大表示聚类结果越好。

3. 模型的构建与求解

3.1. PLID 模型的构建

本文将原始数据 X 分为干净数据、结构化噪声和残差部分,分别用 $D_c L_c$, $D_s L_s$ 和 E 表示。其中 D_c

和 L_c 分别为干净数据的字典和系数矩阵, D_s 和 L_s 分别为结构噪声的字典和系数矩阵, 剩余残差 $E = X - D_c L_c - D_s L_s$ 。A 是将观测矩阵 X 转化为干净数据的映射矩阵, 为此我们提出如下模型:

$$\begin{aligned} \min_{A_s, A_s, A, E} & \|D_c L_c\|_* + \lambda_1 \|D_s L_s\|_1 + \lambda_2 (\lambda_3 \|D_c L_c - AX\|_1 + \|A\|_*) + \frac{1}{2} \|E\|_F^2 \\ \text{s.t.} & X = D_c L_c + D_s L_s + E \end{aligned} \quad (3.1)$$

上式对于任意固定的字典 D_s , 有 $\|D_s L_s\|_1 \leq \|D_s\|_1 \|L_s\|_1$, 所以系数矩阵可以直接控制稀疏性[13]。因此 $D_s L_s$ 和 $D_c L_c$ 的稀疏性和低秩性分别由系数矩阵 L_s 和 L_c 控制。为此将(3.1)式进一步调整后如下:

$$\begin{aligned} \min_{A_s, A_s, A, E} & \|L_c\|_* + \lambda_1 \|L_s\|_1 + \lambda_2 (\lambda_3 \|D_c L_c - AX\|_1 + \|A\|_*) + \frac{1}{2} \|E\|_F^2 \\ \text{s.t.} & X = D_c L_c + D_s L_s + E \end{aligned} \quad (3.2)$$

3.2. PLID 模型的优化求解

对于模型(3.2)中的变量, 本文利用对称交替方向乘子法(ADMM)解决该最小化问题[14]。首先, 引入辅助变量 Q, P, M, N 得到如下的优化问题:

$$\begin{aligned} \min_{A_s, A_s, A, P, Q, M, N, E} & \|Q\|_* + \lambda_1 \|P\|_1 + \lambda_2 (\lambda_3 \|M\|_1 + \|N\|_*) + \frac{1}{2} \|E\|_F^2 \\ \text{s.t.} & L_c = Q, L_s = P, D_c L_c - AX = M, \\ & A = N, X = D_c L_c + D_s L_s + E \end{aligned} \quad (3.3)$$

问题(3.3)对应的增广拉格朗日函数为

$$\begin{aligned} & L_\mu(L_c, L_s, A, P, Q, M, N, E) \\ & = \arg \min_{A_s, A_s, A, P, Q, M, N, E} \|Q\|_* + \lambda_1 \|P\|_1 + \lambda_2 (\lambda_3 \|M\|_1 + \|N\|_*) + \frac{1}{2} \|E\|_F^2 \\ & \quad + \text{Tr}(Y_1^T (L_c - Q)) + \frac{\mu}{2} \|L_c - Q\|_F^2 + \text{Tr}(Y_2^T (L_s - P)) + \frac{\mu}{2} \|L_s - P\|_F^2 \\ & \quad + \text{Tr}(Y_3^T (D_c L_c - AX - M)) + \frac{\mu}{2} \|D_c L_c - AX - M\|_F^2 + \text{Tr}(Y_4^T (A - N)) \\ & \quad + \frac{\mu}{2} \|A - N\|_F^2 + \text{Tr}(Y_5^T (X - D_c L_c - D_s L_s - E)) + \frac{\mu}{2} \|X - D_c L_c - D_s L_s - E\|_F^2 \end{aligned} \quad (3.4)$$

其中 Y_1, Y_2, Y_3, Y_4, Y_5 是拉格朗日乘子, $\mu > 0$ 是罚参数。

1) 求解 Q 的子问题, 保持其他变量不变, 更新 Q :

$$\begin{aligned} Q^* & = \arg \min_Q \|Q\|_* + \text{Tr}(Y_1^T (L_c - Q)) + \frac{\mu}{2} \|L_c - Q\|_F^2 \\ & = \arg \min_Q \frac{1}{\mu} \|Q\|_* + \frac{1}{2} \left\| Q - \left(L_c + \frac{Y_1}{\mu} \right) \right\|_F^2 \\ & = \Omega_{1/\mu}(H) \end{aligned} \quad (3.5)$$

其中, $H = L_c + \frac{Y_1}{\mu}$, $\Omega_{1/\mu}(H) = U \Psi_{1/\mu}[\Sigma] V^T$ 表示奇异值阈值算子[15], $\Psi_\varepsilon[x] = \text{sgn}(x) \max(|x| - \varepsilon, 0)$ 是收缩算子, $U[\Sigma]V$ 是 H 的奇异值分解。

2) 求解 P 的子问题, 保持其他变量不变, 更新 P :

$$\begin{aligned}
P^* &= \arg \min_P \lambda_1 \|P\|_1 + Tr(Y_2^T (L_s - P)) + \frac{\mu}{2} \|L_s - P\|_F^2 \\
&= \arg \min_P \frac{\lambda_1}{\mu} \|P\|_1 + \frac{1}{2} \left\| P - \left(L_s + \frac{Y_2}{\mu} \right) \right\|_F^2 \\
&= \Psi_{\lambda_1/\mu}(\Gamma)
\end{aligned} \tag{3.6}$$

其中, $\Gamma = L_s + \frac{Y_2}{\mu}$ 。

3) 求解 M 的子问题, 保持其他变量不变, 更新 M :

$$\begin{aligned}
M^* &= \arg \min_M \lambda_1 \lambda_2 \|M\|_1 + Tr(Y_3^T (D_c L_c - AX - M)) + \frac{\mu}{2} \|D_c L_c - AX - M\|_F^2 \\
&= \arg \min_M \lambda_1 \lambda_2 \|M\|_1 + \frac{\mu}{2} \left\| M - \left(D_c L_c - AX + \frac{Y_3}{\mu} \right) \right\|_F^2 \\
&= \Psi_{\lambda_2 \lambda_3 / \mu}(Z)
\end{aligned} \tag{3.7}$$

其中, $Z = D_c L_c - AX + \frac{Y_3}{\mu}$ 。

4) 求解 N 的子问题, 保持其他变量不变, 更新 N :

$$\begin{aligned}
N^* &= \arg \min_N \lambda_2 \|N\|_* + Tr(Y_4^T (A - N)) + \frac{\mu}{2} \|A - N\|_F^2 \\
&= \arg \min_N \lambda_2 \|N\|_* + \frac{\mu}{2} \left\| N - \left(A + \frac{Y_4}{\mu} \right) \right\|_F^2 \\
&= \Omega_{\lambda_2/\mu}(K)
\end{aligned} \tag{3.8}$$

其中, $K = A + \frac{Y_4}{\mu}$, $\Omega_{\lambda_2/\mu}(K) = U \Psi_{\lambda_2/\mu}[\Sigma] V^T$ 是奇异值阈值算子。

5) 求解 L_c 的子问题, 保持其他变量不变, 更新 L_c :

$$\begin{aligned}
L_c^* &= \arg \min_{L_c} Tr(Y_1^T (L_c - Q)) + \frac{\mu}{2} \|L_c - Q\|_F^2 \\
&\quad + Tr(Y_3^T (D_c L_c - AX - M)) + \frac{\mu}{2} \|D_c L_c - AX - M\|_F^2 \\
&\quad + Tr\left(Y_5^T (X - D_c L_c - D_s L_s - E) + \frac{\mu}{2} \|X - D_c L_c - D_s L_s - E\|_F^2\right)
\end{aligned} \tag{3.9}$$

该子问题的最优解为:

$$L_c^* = (I + 2D_c^T D_c)^{-1} K_1 \tag{3.10}$$

其中, $K_1 = Q + D_c^T (AX + M - D_s L_s - E + X) - \mu^{-1} (Y_1 + D_c^T Y_3 - D_c^T Y_5)$ 。

6) 求解 L_s 的子问题, 保持其他变量不变, 更新 L_s :

$$\begin{aligned}
L_s^* &= \arg \min_{L_s} Tr(Y_2^T (L_s - P)) + \frac{\mu}{2} \|L_s - P\|_F^2 \\
&\quad + Tr\left(Y_5^T (X - D_c L_c - D_s L_s - E) + \frac{\mu}{2} \|X - D_c L_c - D_s L_s - E\|_F^2\right)
\end{aligned} \tag{3.11}$$

该子问题的最优解为:

$$L_s^* = (I + D_s^T D_s)^{-1} K_2 \quad (3.12)$$

其中, $K_2 = P + D_s^T (X - D_c L_c - E) - \mu^{-1} (Y_2 - D_s^T Y_5)$ 。

7) 求解 A 的子问题, 保持其他变量不变, 更新 A :

$$\begin{aligned} A^* = \arg \min_A & Tr(Y_3^T (D_c L_c - AX - M)) + \frac{\mu}{2} \|D_c L_c - AX - M\|_F^2 \\ & + Tr(Y_4^T (A - N)) + \frac{\mu}{2} \|A - N\|_F^2 \end{aligned} \quad (3.13)$$

该子问题的最优解为:

$$A^* = K_3 (I + XX^T)^{-1} \quad (3.14)$$

其中, $K_3 = D_c L_c X^T - MX^T + N + \mu^{-1} (Y_3 X^T - Y_4)$ 。

8) 求解 E 的子问题, 保持其他变量不变, 更新 E :

$$\begin{aligned} E^* = \arg \min_E & \frac{1}{2} \|E\|_F^2 + Tr(Y_5^T (X - D_c L_c - D_s L_s - E)) \\ & + \frac{\mu}{2} \|X - D_c L_c - D_s L_s - E\|_F^2 \end{aligned} \quad (3.15)$$

该子问题的最优解为:

$$E^* = \frac{\mu}{\mu+1} \left(X - D_c L_c - D_s L_s + \frac{Y_5}{\mu} \right) \quad (3.16)$$

算法 1 给出了求解优化问题(3.3)的 PLID 迭代过程

算法 1. PLID 迭代过程

输入: 数据矩阵 X , 字典 D_c , D_s , 参数 λ_1 , λ_2 , λ_3 ;

初始化: $L_c = L_s = M = Y_1 = Y_2 = Y_3 = 0, \mu = 0.1$,

$\mu_{\max} = 10^6, A = N = Y_4 = Y_5 = 0, \rho = 1.1, \varepsilon = 10^{-6}$

while 不满足收敛条件 **do**

1) 更新拉格朗日乘子中间步

$$Y_1^{k+\frac{1}{2}} = Y_1^k + \mu(L_c^k - Q^k)$$

$$Y_2^{k+\frac{1}{2}} = Y_2^k + \mu(L_s^k - P^k)$$

$$Y_3^{k+\frac{1}{2}} = Y_3^k + \mu(D_c L_c^k - A^k X - M^k)$$

$$Y_4^{k+\frac{1}{2}} = Y_4^k + \mu(A^k - N^k)$$

$$Y_5^{k+\frac{1}{2}} = Y_5^k + \mu(X - D_c L_c^k - D_s L_s^k - E^k)$$

2) 通过公式(3.5), (3.6), (3.7), (3.8), (3.10), (3.12), (3.14), (3.16)更新 $Q^{k+1}, P^{k+1}, M^{k+1}, N^{k+1}, A_c^{k+1}, A_s^{k+1}, A^{k+1}, E^{k+1}$ 。

3) 更新拉格朗日乘子步和参数 μ

续表

$$\begin{aligned}
Y_1^{k+1} &= Y_1^{k+\frac{1}{2}} + \mu(L_c^{k+1} - Q^{k+1}) \\
Y_2^{k+1} &= Y_2^{k+\frac{1}{2}} + \mu(L_s^{k+1} - P^{k+1}) \\
Y_3^{k+1} &= Y_3^{k+\frac{1}{2}} + \mu(D_c L_c^{k+1} - A^{k+1} X - M^{k+1}) \\
Y_4^{k+1} &= Y_4^{k+\frac{1}{2}} + \mu(A^{k+1} - N^{k+1}) \\
Y_5^{k+1} &= Y_5^{k+\frac{1}{2}} + \mu(X - D_c L_c^{k+1} - D_s L_s^{k+1} - E^{k+1}) \\
\mu^* &= \min(\rho * \mu, \mu^{\max})
\end{aligned}$$

4) 检查收敛条件

$$\begin{aligned}
\|L_c^{k+1} - Q^{k+1}\|_\infty &< \varepsilon, \|L_s^{k+1} - P^{k+1}\|_\infty < \varepsilon, \\
\|D_c L_c^{k+1} - A^{k+1} X - M^{k+1}\|_\infty &< \varepsilon, \|A^{k+1} - N^{k+1}\|_\infty < \varepsilon, \\
\|X - D_c L_c^{k+1} - D_s L_s^{k+1} - E^{k+1}\|_\infty &< \varepsilon,
\end{aligned}$$

end while

输出: L_c^{k+1} , L_s^{k+1} , A^{k+1} , E^{k+1}

3.3. 算法的复杂度分析

算法 1 中, Q 和 N 涉及了对矩阵进行奇异值分解, 所以求解 L_c 和 A 的步骤是最耗时的。对于一个矩阵 $X \in R^{m \times n}$, 精确 SVD 的计算复杂度为 $O(\min(mn^2, m^2n))$ 。因此, 更新 $Q \in R^{d_c \times n}$ 的计算复杂度为 $O(\min(d_c n^2, d_c^2 n))$, 类似的更新 $N \in R^{m \times m}$ 的计算复杂度为 $O(m^3)$ 。

4. 实验结果与讨论

在本节中, 首先在 ORL¹ 和 Yale² 数据集上进行消融实验, 验证 PLID 算法各个成分的有效性。其次对 PLID 算法的收敛性进行分析。最后在 AR 数据集[16]上与 RPCA [10], PSSV [17], DRPCA [18], ARPCA [19]等算法进行比较。表 1 总结了实验所用的数据集的特征。

Table 1. Description of the experimental dataset
表 1. 实验数据集的描述

Type	Dataset	Class	Number	Size
Face	ORL	40	400	28 × 23
	Yale	15	165	25 × 25
	AR	100	2600	55 × 40

4.1. 预学习字典

在算法 1 的框架之外, 需要提前学习字典 D_c 和 D_s 。首先采用 RPCA 方法将原始数据分解为干净部分和结构化噪声, 然后利用 KSVD 学习这两部分的字典 D_c 和 D_s , 两个字典分别为干净数据字典和结构化噪声字典。本文对 ORL, AR 和 Yale 数据集干净部分和结构化噪声的两个字典的原子数分别设置为 2 和 3。

4.2. 收敛性分析

当涉及到具有三个以上变量块的凸问题时, ADMM 可能是发散的[20], 因此本文采用实验证明算法的收敛性, 其中 $\|r^k\| = \max \left\{ \begin{array}{l} \|L_c^{k+1} - Q^{k+1}\|_\infty, \|L_s^{k+1} - P^{k+1}\|_\infty, \|D_c L_c^{k+1} - A^{k+1} X - M^{k+1}\|_\infty, \\ \|A^{k+1} - N^{k+1}\|_\infty, \|X - D_c L_c^{k+1} - D_s L_s^{k+1} - E^{k+1}\|_\infty \end{array} \right\} < \varepsilon$ 从图 1 可以看出, PLID 在所有数据集中大约会在迭代 80 次收敛。

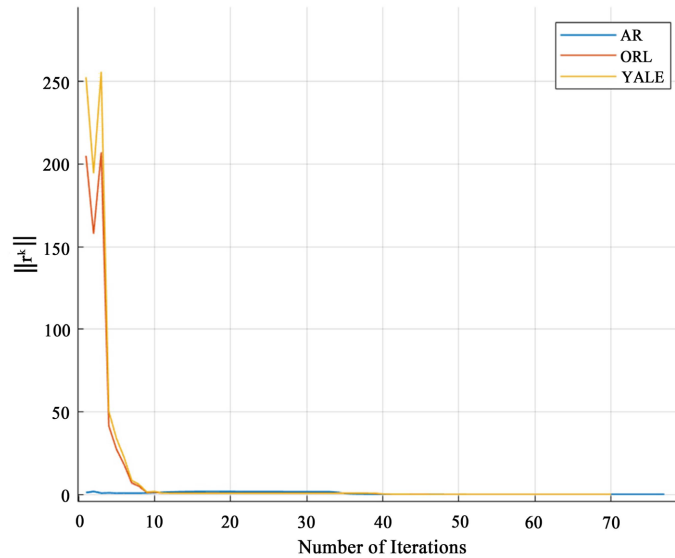


Figure 1. Convergence curves of PLID on three datasets

图 1. PLID 在三个数据集的收敛曲线

4.3. 消融实验

上述所提出的模型是由几个模块组成, 为验证各个模块的相关性和有效性, 我们对其进行消融实验, 具体方式如下:

- 1) 为了评估预学习字典的作用, 将 PLID 的完整版本与没有预学习字典的方法进行比较, 即干净数据和结构化噪声不分解为字典和系数矩阵的乘积。
- 2) 理论上, 对不同的数据分别进行预学习字典可以提高算法的恢复性能。为了证实这种潜在的优势, 将完整版本的算法和它的简化版本进行比较, 即字典用单位矩阵表示。
- 3) 为了验证正则项的作用, 对 PLID 在参数 λ_2 为零和非零时的性能进行比较。

最后, 用准确度(Accuracy, ACC), 归一化互信息(normalized mutual information, NMI)和纯度(Purity, PUR)做为评价指标, 对 ORL 和 Yale 数据集进行聚类实验, 检验各个组件的聚类效果, 所得结果汇总于表 2。表中结果表明, PLID 的每个模块在提高算法的性能方面发挥着积极和显著的作用。

4.4. 人脸去噪实验

首先随机在 AR 数据集上[16]选择 5 个类, 然后进一步选择每类中 14 张无遮挡的干净图像和 3 张带眼镜的图像作为训练数据, 剩下 3 张带眼镜图像作为每类的测试数据。将每张图片的大小重新调整为 55×40 , 参数设置 $\lambda_1 = 0.01$, $\lambda_2 = 0.5/\sqrt{\max(m, n)}$, $\lambda_3 = 1.8$ 。此外, 我们将同时对每张图像添加如下的两种噪声:

Table 2. Cluster index analysis of different components of PLID
表 2. PLID 不同组成成分的聚类指标分析

	指标	没有预学习字典	字典矩阵为单位矩阵	$\lambda_2 = 0$	PLID
ORL	ACC	0.3846	0.8462	0.7308	0.8846
	NMI	0.5313	0.8272	0.8109	0.8583
	PUR	0.5385	0.8462	0.8077	0.8846
Yale	ACC	0.3846	0.3462	0.7692	0.8077
	NMI	0.2680	0.1796	0.7062	0.7318
	PUR	0.4615	0.3462	0.7692	0.8077

- 1) 添加信噪比为 15 dB 的高斯噪声
- 2) 添加 3 个大小为 5×5 黑色遮挡块

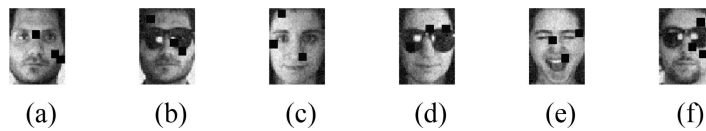


Figure 2. Example of AR image corruption
图 2. AR 图像损坏示例

上图 2 是添加两种噪声后的图像示例。下面将提出的算法与其他方法进行比较，并使用 k-means 方法对获得的干净数据进行聚类分析，聚类结果如表 3 所示。

Table 3. Clustering results of corrupted AR images
表 3. 损坏的 AR 图像聚类结果

指标	RPCA	PSSV	DRPCA	ARPCA	SLRPD
ACC	0.7294	0.5412	0.5529	0.5647	0.8118
NMI	0.6849	0.5703	0.3763	0.6101	0.7611
PUR	0.7882	0.5647	0.5882	0.6353	0.8471

从表 3 可以看出，PLID 相比其它四种算法对损坏图像的聚类效果更好，体现了更好的性能。同时，图 3 展示了不同算法的恢复图像，很容易发现 RPCA, PSSV, DRPCA 和 ARPCA 算法可以很好的处理一部分高斯噪声和黑色遮挡块，但是不能很好地处理人脸的眼镜遮挡。相比之下，本文所提的算法不仅去除了眼镜干扰，还很清晰地去掉了三个黑色块，说明 PLID 更适合处理复杂的噪声情况。

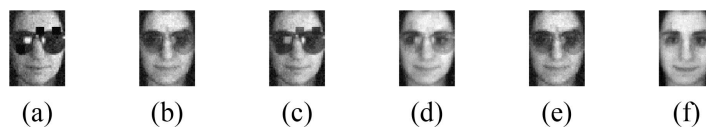


Figure 3. Example of recovery. (a) is the image damaged, (b)~(f) is the image reconstructed by RPCA, PSSV, DWRPCA, ARPCA, and PLID, respectively
图 3. 算法恢复 AR 图像的可视化示例。(a) 是被噪声损坏的图像，(b)~(f) 分别是被 RPCA, PSSV, DWRPCA, ARPCA 和 PLID 重建的图像

此外, 为了验证映射矩阵 A 对新样本(即测试数据)的有效性, 先使用 PLID 通过训练数据求得 A , 然后对测试数据中 5 个类的 10 张带有太阳眼镜图像直接处理, 恢复结果如图 4 所示。可以看出, 添加噪声的测试样本在通过映射矩阵的线性变换后, 也可以恢复出一张清晰的人脸。

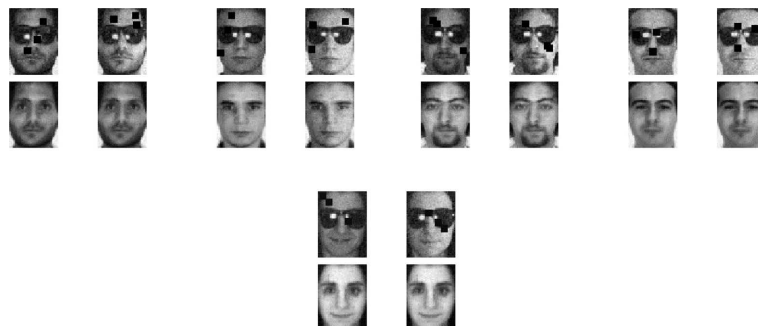


Figure 4. Example of recovery of the test sample. The first row is the test sample with added noise, and the second row is the data AX reconstructed with mapping matrix A

图 4. 测试样本的恢复示例。第一行是添加噪声的测试样本, 第二行是利用映射矩阵 A 重构的数据 AX

5. 结论

本文提出了一种基于预学习字典的低秩算法, 该算法将干净数据和结构化噪声分别表示为字典矩阵和系数矩阵的乘积, 同时在重构误差和低秩结构之间建立联系。预学习的字典可以表示小数据集中的子空间, 可以提高图像的去噪性能。此外, 基于训练数据得到的映射矩阵可以用于处理未参与训练的新样本。采用对称 ADMM 将优化模型的目标函数分解为若干个子问题进行求解, 大大减少了数据集中恢复出干净数据的计算量并加快了算法的迭代速度。最后在 AR 数据集上的实验表明, 该方法从噪声数据中恢复具有低秩结构的干净数据方面优于其他方法。

参考文献

- [1] Chen, R., Pu, D., Tong, Y., *et al.* (2022) Image-Denoising Algorithm Based on Improved K-Singular Value Decomposition and Atom Optimization. *CAAI Transactions on Intelligence Technology*, **7**, 117-127. <https://doi.org/10.1049/cit2.12044>
- [2] Sahoo, S.K. and Makur, A. (2015) Replacing K-SVD with SGK: Dictionary Training for Sparse Representation of Images. 2015 *IEEE International Conference on Digital Signal Processing*, Singapore, 21-24 July 2015, 614-617. <https://doi.org/10.1109/ICDSP.2015.7251947>
- [3] Song, X., Liu, Z., Yang, X. and Yang, J. (2014) A Parameterized Fuzzy Adaptive K-SVD Approach for the Multi-Class Study of Pursuit Algorithms. *Neurocomputing*, **123**, 131-139. <https://doi.org/10.1016/j.neucom.2013.06.017>
- [4] Liu, G., Lin, Z., Yan, S., *et al.* (2012) Robust Recovery of Subspace Structures by Low-Rank Representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **35**, 171-184. <https://doi.org/10.1109/TPAMI.2012.88>
- [5] Mairal, J., Bach, F. and Ponce, J. (2011) Task-driven Dictionary Learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **34**, 791-804. <https://doi.org/10.1109/TPAMI.2011.156>
- [6] Chen, Z., Papadopoulos, H. and Ellis, D.P. (2014) Content-Adaptive Speech Enhancement by a Sparsely-Activated Dictionary plus Low Rank Decomposition. 2014 *4th Joint Workshop on Hands-free Speech Communication and Microphone Arrays*, Villers-les-Nancy, 12-14 May 2014, 16-20. <https://doi.org/10.1109/HSCMA.2014.6843242>
- [7] Xu, S., Yang, X. and Jiang, S. (2017) A Fast Nonlocally Centralized Sparse Representation Algorithm for Image Denoising. *Signal Processing*, **131**, 99-112. <https://doi.org/10.1016/j.sigpro.2016.08.006>
- [8] Wright, J., Ma, Y., Mairal, J., *et al.* (2010) Sparse Representation for Computer Vision and Pattern Recognition. *Proceedings of the IEEE*, **98**, 1031-1044. <https://doi.org/10.1109/JPROC.2010.2044470>

-
- [9] Zhou, P., Fang, C., Lin, Z., *et al.* (2018) Dictionary Learning with Structured Noise. *Neurocomputing*, **273**, 414-423. <https://doi.org/10.1016/j.neucom.2017.07.041>
- [10] Candès, E. J., Li, X., Ma, Y. and Wright, J. (2011) Robust Principal Component Analysis? *Journal of the ACM*, **58**, 1-37. <https://doi.org/10.1145/1970392.1970395>
- [11] Murphy, K. P. (2012) *Machine Learning: A Probabilistic Perspective*. MIT Press, Cambridge.
- [12] Vinh, N. X., Epps, J. and Bailey, J. (2010) Information Theoretic Measures for Clusterings Comparison: Variants, Properties, Normalization and Correction for Chance. *Journal of Machine Learning Research*, **11**, 2837-2854.
- [13] Yu, S., Zhang, H. and Duan, Z. (2017) Singing Voice Separation by Low-Rank and Sparse Spectrogram Decomposition with Prelearned Dictionaries. *Journal of the Audio Engineering Society*, **65**, 377-388. <https://doi.org/10.17743/jaes.2017.0009>
- [14] Chao, M.T., Han, D. R. and Cai, X.J. (2021) Convergence of the Peaceman-Rachford Splitting Method for a Class of Nonconvex Programs. *Numerical Mathematics Theory Methods and Applications*, **14**, 438-460. <https://doi.org/10.4208/nmtma.OA-2020-0063>
- [15] Cai, J.F., Candès, E.J. and Shen, Z. (2010) A Singular Value Thresholding Algorithm for Matrix Completion. *SIAM Journal on Optimization*, **20**, 1956-1982. <https://doi.org/10.1137/080738970>
- [16] Martinez, A. and Benavente, R. (1998) The AR Face Database. *CVC Technical Report*, **24**, 1-10.
- [17] Oh, T.H., Tai, Y.W., Bazin, J.C., *et al.* (2015) Partial Sum Minimization of Singular Values in Robust PCA: Algorithm and Applications. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **38**, 744-758. <https://doi.org/10.1109/TPAMI.2015.2465956>
- [18] Wang, Q., Gao, Q., Sun, G., *et al.* (2020) Double Robust Principal Component Analysis. *Neurocomputing*, **391**, 119-128. <https://doi.org/10.1016/j.neucom.2020.01.097>
- [19] Liu, Y., Gao, X., Gao, Q., *et al.* (2019) Adaptive Robust Principal Component Analysis. *Neural Networks*, **119**, 85-92. <https://doi.org/10.1016/j.neunet.2019.07.015>
- [20] Chen, C., He, B., Ye, Y. and Yuan, X. (2016) The Direct Extension of ADMM for Multi-B-lock Convex Minimization Problems Is Not Necessarily Convergent. *Mathematical Programming*, **155**, 57-79. <https://doi.org/10.1007/s10107-014-0826-5>