

自适应匹配追踪图像去噪算法

李桂会 李晋江 范辉

山东工商学院计算机科学与技术学院 山东烟台 264000

山东省高等学校协同创新中心:未来智能计算 山东烟台 264000

(guihui@163.com)



摘要 针对目前的稀疏去噪算法分解效率低、去噪效果不理想的问题,提出了一种基于自适应匹配追踪的图像去噪算法。该算法首先通过自适应匹配追踪算法求解稀疏系数,然后利用K奇异值分解算法将字典训练成能够有效反映图像结构特征的自适应字典,最后将稀疏系数与自适应字典相结合来重构图像。在重构过程中,将噪声对应的系数去除,最终达到去噪的效果。算法引入Spike-Slab先验来引导稀疏系数矩阵的稀疏性,并利用两个权重矩阵促使去噪模型更加真实。鉴于字典在稀疏算法中的重要性,将自适应字典与DCT冗余字典、Global字典进行比较。实验结果显示,选择自适应字典的去噪结果比传统字典在峰值信噪比上高出约4.5 dB;与目前6种主流的稀疏去噪方法相比,文中提出的方法在3种评价指标上均有不同程度的提高,其中峰值信噪比平均提高了约0.76~6.24 dB,特征相似度平均提高了约0.012~0.082,结构相似度平均提高了约0.015~0.108。对图像去噪算法进行定性的评价,结果显示所提算法保留了更多的有用信息,视觉效果最佳。实验充分证明了自适应匹配追踪图像去噪算法对图像去噪的有效性和鲁棒性。

关键词: 图像去噪;稀疏表示;自适应匹配追踪;K奇异值分解;Spike-Slab先验

中图分类号 TP391.41

Image Denoising Algorithm Based on Adaptive Matching Pursuit

LI Gui-hui, LI Jin-jiang and FAN Hui

School of Computer Science and Technology, Shandong Technology and Business University, Yantai, Shandong 264000, China

Co-innovation Center of Shandong Colleges and Universities; Future Intelligent Computing, Yantai, Shandong 264000, China

Abstract Aiming at the problem that the current sparse denoising algorithm has low decomposition efficiency and unsatisfactory denoising effect, an image denoising algorithm based on adaptive matching pursuit was proposed. Firstly, the algorithm uses the adaptive matching pursuit algorithm to solve the sparse coefficients, and then uses the K-means singular value decomposition algorithm to train the dictionary into an adaptive dictionary that can effectively reflect the image structure features. Finally, the image is reconstructed by combining the sparse coefficient with the adaptive dictionary. During the reconstruction process, the coefficients corresponding to the noise are removed, and finally the denoising effect is achieved. Spike-Slab priori is introduced to guide the sparsity of sparse coefficient matrix, and two weight matrices are used to make the denoising model more realistic. In view of the importance of dictionary in sparse algorithm, this paper compared adaptive dictionary with DCT redundant dictionary and Global dictionary. The experimental results show that the denoising result of adaptive dictionary is about 4.5 dB higher than that of traditional dictionary in terms of peak signal-to-noise ratio (PSNR). The proposed method improves three evaluation indicators in varying degrees compared with the current six main methods of sparse denoising. The PSNR is increased by about 0.76 dB to 6.24 dB, the feature similarity (FSIM) is increased by about 0.012 to 0.082, and the structure similarity (SSIM) is increased by about 0.015 to 0.108 on average. The qualitative evaluation of the image denoising algorithm shows that the proposed algorithm retains more useful information and has the best visual effect. Therefore, the experiment fully proves its effectiveness and robustness.

Keywords Image denoising, Sparse representation, Adaptive matching pursuit, K-means singular value decomposition, Spike-Slab priori

收稿日期: 2018-12-09 返修日期: 2019-04-12 本文已加入开放科学计划(OSID),请扫描上方二维码获取补充信息。

基金项目:国家自然科学基金(61472227,61772319,61602277)

This work was supported by the National Natural Science Foundation of China (61472227,61772319,61602277).

通信作者:李晋江(lijinjiang@gmail.com)

1 引言

图像噪声是图像在摄取或传输过程中所受到的随机信号干扰造成,这种干扰不仅影响了图像的视觉效果,而且阻碍了人们对正确信息的识别。数学上,图像去噪模型可表示为:

$$\mathbf{X}=\mathbf{Y}+\varepsilon \quad (1)$$

其中, \mathbf{Y} 是干净的图像, ε 是高斯白噪声, \mathbf{X} 表示含噪图像。图像去噪过程的实质是从 \mathbf{X} 中恢复 \mathbf{Y} 的过程。现实生活中,许多工业和科学领域都要求使用清晰的图像,但是实际得到的图像存在较严重的噪声干扰,因此图像去噪是一个极具应用意义的研究课题。

很多研究人员致力于图像去噪的研究,并取得了丰富的成果。他们努力寻找合适的去噪方法,以改善图像的质量。目前常用的图像去噪方法大致可以分为空域滤波^[1]、变换域滤波^[2-3]、混合滤波^[4-5]3种。以上方法虽然都能起到抑制图像噪声的作用,但是它们的去噪结果均存在较严重的图像细节缺失现象,导致图像变得不清晰。近年来,在图像处理研究中,稀疏表示算法因具有良好的去噪效果而得到了广泛的关注和研究。与传统的去噪算法相比,基于稀疏表示的图像去噪算法能够灵活地表达图像的多种结构,且数据处理成本低。稀疏去噪模型首先通过字典训练算法训练出自适应字典,然后利用稀疏分解算法求得稀疏系数,最后将两者结合以重构无噪图像。因此,稀疏去噪模型解决的两个关键问题是自适应字典的训练和稀疏系数的求解。

自适应字典的训练^[6-9]是稀疏分解的保证,利用它完全可以线性表示初始字典的信息。它保留了图像的多种结构特征,能充分地表示图像信息,且比传统的正交基字典更加注重图像表示的简易性和高效性。为了进一步对字典进行优化,Engan等^[10]提出了最优方向法(Method of Optimal Directions,MOD),它的核心思想是根据表示向量的有限数量的非零分量来解决最优化问题。然而,MOD算法中矩阵求逆操作的高度复杂性促进了其他字典学习的发展。随后,Ahroni等^[11]提出了KSVD算法,该算法被认为是字典学习的标准,并在实际中得到广泛应用。KSVD算法充分利用了稀疏表示的优势,大大减少了原子的数量,降低了计算的复杂度,提高了数据的处理速度。

稀疏分解是图像稀疏表示的另一个关键问题。Mallat等^[12]提出了图像稀疏分解的匹配追踪MP方法,该方法在完备字典库上对信号进行分解,所以存在已入选的原子可能再次被选中的缺点,从而降低了算法效率。对于MP算法中存在的一系列问题,正交匹配追踪(Orthogonal Matching Pursuit,OMP)算法^[13]、分段匹配追踪(Stage-wise Orthogonal Matching Pursuit,StOMP)算法^[14]、梯度匹配追踪(Gradient Matching Pursuit,GMP)算法^[15]、最速下降正交匹配追踪(Steepest descent Orthogonal Matching Pursuit,SdOMP)算法^[16]、随机细化正交匹配追踪(Random Refined Orthogonal Matching Pursuit,RROMP)算法^[17]、稀疏度自适应匹配追踪(Sparsity Adaptive Matching Pursuit,SAMP)算法^[18]等一系列贪婪算法相继被提出,同时还出现了基于贝叶斯的方法^[19-20]或一般的稀疏近似算法。但是,目前的算法均存在一定的缺陷,如鲁棒性差、重建效果不够理想、计算成本昂贵、涉及简化或松弛等。

随着一种新的自适应匹配追踪^[21]算法被提出,稀疏系数带来的硬非凸和混合整数规划的难题得到了解决。本文在稀疏表示理论与自适应匹配追踪算法的基础上,将图像去噪转化为一个最优化问题,构建了基于先验和稀疏表示的图像去噪模型。算法流程是:首先,利用字典学习算法将图像进行稀疏表示;然后,用稀疏分解算法和Spike-Slab先验构造出稀疏模型,获得稀疏系数;最后,将自适应字典与稀疏系数结合,重构出干净的图像,从而达到去噪的目的。将本文算法与多种去噪算法进行比较,实验结果显示,本文算法较其他方法在主观评价和客观评价上均具有优势。

2 自适应匹配追踪

稀疏表示图像的任务可以用公式表示为:

$$\min \|\mathbf{A}\|_0, \mathbf{X}=\mathbf{B}\mathbf{A} \quad (2)$$

即找到一个满足 $\mathbf{X}=\mathbf{B}\mathbf{A}$ 的稀疏矩阵 \mathbf{A} 。在文献^[21]的基础上,本文利用自适应匹配追踪算法求解稀疏分解问题。算法的本质是通过向索引集中添加或删除元素来更新索引,进而通过更新的索引集求得稀疏系数的最优解。为了保证算法的效率,通过廉价Cholesky分解^[22]来计算算法的中间步骤。文献^[23-26]已经表明,向框架中引入先验分布能够保证稀疏性。本文使用Spike和Slab先验^[27]来引导稀疏矩阵上的稀疏性,其中每个系数 a_i 代表了两种密度,如下所示:

$$a_i \sim (1-\beta_i)\Pi(a_i=0)+\beta_i\mathcal{Y}_i(a_i) \quad (3)$$

其中, $\Pi(\cdot)$ 是在零Spike处的指标函数;对于高斯分布 a_i , \mathcal{Y}_i 是一个用于非零值Slab的先验分布; $\beta_i \in \{0,1\}$ 控制 a_i 的活跃度。从贝叶斯观点来看,式(2)的优化问题可以通过最大化后验概率来获得解:

$$\begin{aligned} \mathbf{A}^* &= \operatorname{argmin} \left\{ \|\mathbf{X}-\mathbf{B}\mathbf{A}\|_2^2 + \lambda \|\mathbf{A}\|_2^2 + \sum_{i=1}^M \varphi_i \gamma_i \right\} \\ \varphi_i &\stackrel{\Delta}{=} \sigma^2 \log \left(\frac{2\pi\sigma^2(1-\eta_i)^2}{\lambda\eta_i^2} \right) \end{aligned} \quad (4)$$

其中, λ 表示正则化参数; σ^2 表示高斯分布的方差; γ 代表 \mathbf{A} 的指示变量,如果 \mathbf{A} 中元素为0,则 $\gamma=0$,否则 $\gamma=1$; η_i 通过分别控制每个指标的 γ 是否有效来影响稀疏性。如果 η_i 足够大, φ_i 可能是负数。

为了解释图像噪声的变化统计,引入两个权重矩阵 \mathbf{T}_1 和 \mathbf{T}_2 。权重矩阵 \mathbf{T}_1 和 \mathbf{T}_2 采用最大后验(MAP)估计技术进行确定。具体而言,根据局部块中的逼真噪声可近似建模为高斯噪声的经验,并且每个局部块具有高斯分布,可以得到:

$$P(\mathbf{X}|\mathbf{A}) = \prod_{c \in \{r,g,b\}} \prod_{k=1}^n (\pi\sigma_{c,k})^{-p^2} e^{-\sigma_{c,k}^{-2} \|\mathbf{x}_{c,k} - \mathbf{b}_c \mathbf{a}_k\|} \quad (5)$$

其中, $\mathbf{x}_{c,k}$ 和 \mathbf{a}_k 分别表示矩阵 \mathbf{Y}_c 和 \mathbf{A} 的第 k 列, $\sigma_{c,k}$ 表示 x_c 的噪声标准。 $\sigma_{c,k}$ 可被看作由 σ_c 和 σ_m 的列联表组成,因此 $\sigma_{c,k}$ 可以表示为: $\sigma_{c,k} = \sigma^{\theta_1} \sigma^{\theta_2}$, $\theta_1 + \theta_2 = 1$ 。本文设定 $\theta_1 = \frac{1}{2}$, $\theta_2 =$

$\frac{1}{2}$,将 $\sigma_{c,k} = \sigma^{\frac{1}{2}} \sigma^{\frac{1}{2}}$ 代入式(4),可以推导出:

$$\begin{aligned} \mathbf{A}^* &= \operatorname{argmin} \left\{ \sum_{c,k=1}^n \sigma_{c,k}^{-2} \|\mathbf{x}_{c,k} - \mathbf{b}_c \mathbf{a}_k\| + \lambda \|\mathbf{A}\|_2^2 + \sum_{i=1}^M \varphi_i \gamma_i \right\} \\ &= \operatorname{argmin} \left\{ \sum_c \sigma_c^{-1} \|(\mathbf{X}_c - \mathbf{B}\mathbf{A})\mathbf{T}_2\| + \lambda \|\mathbf{A}\|_2^2 + \sum_{i=1}^M \varphi_i \gamma_i \right\} \end{aligned}$$

$$= \operatorname{argmin}\{\|T_1(\mathbf{X}_c - \mathbf{B}\mathbf{A})T_2\| + \lambda\|\mathbf{A}\|_2^2 + \sum_{i=1}^M \varphi_i \gamma_i\}$$

$$T_1 = \operatorname{diag}(\delta_r^{-\frac{1}{2}}\mathbf{I}, \delta_g^{-\frac{1}{2}}\mathbf{I}, \delta_b^{-\frac{1}{2}}\mathbf{I}), T_2 = \operatorname{diag}(\delta_1^{-\frac{1}{2}}, \dots, \delta_M^{-\frac{1}{2}})$$

进一步将稀疏模型变换为:

$$\mathbf{A}^* = \operatorname{argmin}\{\|T_1(\mathbf{X} - \mathbf{B}\mathbf{A})T_2\|_2^2 + \lambda\|\mathbf{A}\|_2^2 + \sum_{i=1}^M \varphi_i \gamma_i\} \quad (6)$$

其中, $\mathbf{A} \in \mathbf{R}^{n \times m}$ 是稀疏矩阵, $\mathbf{B} \in \mathbf{R}^{m \times n}$ 是训练的字典, $\mathbf{X} \in \mathbf{R}^{m \times m}$ 是重构的稀疏信号。

2.1 优化

$$\text{为了便于对式(6)进行优化,令 } \mathbf{F} = \begin{bmatrix} \mathbf{X} \\ \mathbf{0} \end{bmatrix}, \mathbf{C} = \begin{bmatrix} \mathbf{B} \\ \sqrt{\lambda\mathbf{I}} \end{bmatrix},$$

$\mathbf{I} \in \mathbf{R}^{m^2}, \mathbf{0} \in \mathbf{R}^{m \times 1}$, 则式(6)可以重写为:

$$\mathbf{A}^* = \operatorname{argmin}\|T_1(\mathbf{F} - \mathbf{C}\mathbf{A})T_2\|_2^2 + \sum_{i=1}^M \varphi_i \gamma_i \quad (7)$$

假设 \mathbf{B} 的每一列都有规范 1, 即 $\|b_i\|_2^2 = 1$, 则 $\|c_i\|_2^2 = \|b_i\|_2^2 + \lambda = 1 + \lambda (i = 1, \dots, m)$. 矩阵 \mathbf{C} 是由 \mathbf{B} 和 \mathbf{I} 构造出的, 对于自适应字典 \mathbf{B} , 利用 K 奇异值分解算法 (K-means Singular Value Decomposition, KSVD) 训练出给定矩阵 \mathbf{X} 的自适应字典 (字典的训练过程见 2.2 节), 因此 \mathbf{C} 很容易被表示出来. 另外, 如果知道权重矩阵 T_1 和 T_2 , 以及信号的真实索引 ω , 即 $\omega = \{i\}$, 那么通过式(8)可以找到式(6)的解:

$$\mathbf{A}^* = \operatorname{argmin}\|T_1(\mathbf{F} - \mathbf{C}_\omega \mathbf{A}_\omega)T_2\|_2^2 \quad (8)$$

$$\mathbf{r}_\omega = T_1(\mathbf{F} - \mathbf{C}_\omega \mathbf{A}_\omega)T_2$$

其中, ω 是索引集, \mathbf{C}_ω 是由 ω 索引的列形成的 \mathbf{C} 的子矩阵, \mathbf{A}_ω 是仅包含由 ω 索引的 \mathbf{A} 的有效系数的向量, \mathbf{r}_ω 是由 ω 产生的残差. $\omega \leftrightarrow \mathbf{A}_\omega$ 表示求解式(6)等同于找到有效集合 ω .

2.1.1 设定权重矩阵

将 \mathbf{F} 变换为具有 3 个通道的矩阵 $\mathbf{F} = (f_r, f_g, f_b)$, 从中提取大小为 $p \times p \times 3$ 的局部块并将其拉伸为矢量. 将两个权重矩阵设置为对角矩阵. T_1 表示通道噪声, 用于描述 R, G, B 通道噪声的特性, 由 3 个对角矩阵块组成, 分别对应到 3 个通道. 它用于规范残差矩阵 $(\mathbf{F} - \mathbf{C}_\omega \mathbf{A}_\omega)$ 的行差异, 且 $T_1 = \operatorname{diag}(\delta_r^{-\frac{1}{2}}\mathbf{I}, \delta_g^{-\frac{1}{2}}\mathbf{I}, \delta_b^{-\frac{1}{2}}\mathbf{I})$. T_2 表示局部块噪声, 它的每个对角线元素描述了相应局部块中的噪声方差. 其用于规范残差矩阵 $(\mathbf{F} - \mathbf{C}_\omega \mathbf{A}_\omega)$ 的列差异, 且 $T_2 = \operatorname{diag}(\delta_1^{-\frac{1}{2}}, \dots, \delta_M^{-\frac{1}{2}})$. 本文采用噪声标准偏差来确定相应通道和贴片中的对角线元素, 且噪声越强, 通道和局部块对去噪输出的贡献就越小.

2.1.2 索引集的求解

为了求得索引集, 首先对索引值进行初始化 $\omega_0 = \{i, \varphi_i < 0\}$. 引理 1 给出了初始化的依据.

引理 1 设 $\hat{\omega}$ 是最优索引集合, 若 $\varphi_i < 0$, 则 $i \in \hat{\omega}$.

证明: 假设 $i \notin \hat{\omega}$. 令 $r_{\hat{\omega}} = T_1(\mathbf{F} - \mathbf{C}_{\hat{\omega}} \mathbf{A}_{\hat{\omega}})T_2$, 求得:

$$\begin{aligned} V(\hat{\omega} \cup \{i\}) &\leq \|r_{\hat{\omega}} - a_i c_i\|_2^2 + \varphi_i \\ &= V(\hat{\omega}) + c_i^T c_i a_i^2 - 2r_{\hat{\omega}}^T c_i a_i + \varphi_i \\ &= V(\hat{\omega}) + \|c_i\|_2^2 a_i^2 - 2r_{\hat{\omega}}^T c_i a_i + \varphi_i \\ &= V(\hat{\omega}) + (1 + \lambda) a_i^2 - 2r_{\hat{\omega}}^T c_i a_i + \varphi_i \end{aligned} \quad (9)$$

令 $G(\mathbf{a}) = \|c_i\|_2^2 \mathbf{a}^2 - 2r_{\hat{\omega}}^T c_i \mathbf{a} + \varphi_i$, 观察到 $G(\mathbf{a})$ 是连续的, 并且 $\mathbf{a} \rightarrow \infty$ 时, $\lim G(\mathbf{a}) = +\infty$, $G(0) = \varphi_i < 0$, 且存在 $\bar{a}_i \neq$

0, 使得 $\varphi_i < G(\bar{a}_i) < 0$, 结合式(9)得到: $V(\hat{\omega} \cup \{i\}) \leq V(\hat{\omega}) + G(\bar{a}_i) < V(\hat{\omega})$. 换句话说, $\hat{\omega} \cup \{i\}$ 产生的成本比 $\hat{\omega}$ 低, 这与 $\hat{\omega}$ 是最优的假设相矛盾. 矛盾表明, i 一定属于索引集 $\hat{\omega}$. 根据以上证明过程, 初始化 $\omega_0 = \{i, \varphi_i < 0\}$.

其次, 为了更新索引集并降低成本, 本文算法通过向 ω 添加一个未被选择的索引或删除 ω 中的一个有效元素来更新索引集. 具体来说, 对于给定的 ω , 定义:

$$V(\omega) = \min_{\mathbf{A}_\omega} \|T_1(\mathbf{F} - \mathbf{C}_\omega \mathbf{A}_\omega)T_2\|_2^2 \quad (10)$$

在每次迭代中, 通过式(11)和式(12)来计算两个中间值.

$$K_\omega = \min_{i \notin \omega} V(\omega \cup \{i\}) - V(\omega) \quad (11)$$

$$L_\omega = \min_{i \in \omega} V(\omega \setminus \{j\}) - V(\omega) \quad (12)$$

式(11)、式(12)表明, 如果在索引 ω 中添加一个未被选择的索引 i , 则代价函数下降; 如果删除一个已经选择的索引 j , 代价函数也会下降. 为了显著降低计算成本, 避免计算 $V(\omega \cup \{i\})$ 和 $V(\omega \setminus \{j\})$ 的高代价, 用 K_ω 和 L_ω 的上界 \bar{K}_ω 和 \bar{L}_ω 代替计算 $V(\omega \cup \{i\})$ 和 $V(\omega \setminus \{j\})$ 的值; 然后根据这些近似值获得 ω^{new} . \bar{K}_ω 和 \bar{L}_ω 的计算公式由定理 1 给出.

定理 1 对于任意一个 $i \notin \omega, j \in \omega$, 存在不等式:

$$K_\omega \leq \min_{i \notin \omega} \left\{ \varphi_i - \frac{(r_\omega^T c_i)^2}{1 + \lambda} \right\} \triangleq \bar{K}_\omega \quad (13)$$

$$L_\omega \leq \min_{j \in \omega} \left\{ (1 + \lambda)(a_j^*)^2 + 2c_j^T r_\omega a_j^* - \varphi_j \right\} \triangleq \bar{L}_\omega \quad (14)$$

其中, a_j^* 是在索引 $j \in \omega$ 处的 \mathbf{A}_ω 中的元素.

证明: 对于 $i \notin \omega$, 由引理 1 得:

$$\begin{aligned} V(\omega \cup \{i\}) &\leq V(\omega) + \min\{\varphi_i + (1 + \lambda)a_i^2 - 2r_\omega^T c_i a_i\} \\ &= V(\omega) + \varphi_i - 2r_\omega^T c_i a_i \end{aligned}$$

因此, 不等式(13)成立.

又因为已知 a_j^* 是在索引 $j \in \omega$ 处的 \mathbf{A}_ω 中的元素, 对于任意一个 $j \in \omega$, 求得:

$$\begin{aligned} V(\omega) &= \|r_\omega + c_j a_j^* - c_j a_j^*\|_2^2 + \varphi_j \\ &= \|r_\omega + c_j a_j^*\|_2^2 + (1 + \lambda)(a_j^*)^2 + \varphi_j - 2c_j^T (r_\omega + c_j a_j^*) a_j^* \\ &\geq V(\omega \setminus \{j\}) - (1 + \lambda)(a_j^*)^2 - 2c_j^T r_\omega a_j^* + \varphi_j \end{aligned}$$

结合式(12), 不等式(14)成立. 综上, 定理 1 得证.

根据定理 1, 计算出的 K_ω 和 L_ω 的上界分别是 \bar{K}_ω 和 \bar{L}_ω , 通过比较 \bar{K}_ω 和 \bar{L}_ω 的大小来更新索引集合. 如果 \bar{K}_ω 和 \bar{L}_ω 都大于或等于 0, 停止算法, 索引集合没有得到改进. 否则, 如果 $\bar{K}_\omega < \bar{L}_\omega$, 则向 ω 增加 i ; 如果 $\bar{K}_\omega > \bar{L}_\omega$, 则通过移除 j 来更新 ω . 这个过程保证了每次迭代后代价函数都会下降, 算法在迭代有限次后, 最终停止. 同时从定理 1 可以看出, \bar{K}_ω 和 \bar{L}_ω 需要的计算量较少, 这提高了算法的效率.

2.1.3 稀疏系数的求解

为了得到稀疏系数, 在得到更新的 ω 之后, 使用 $\mathbf{C}_\omega \mathbf{C}_\omega^T$ 的 Cholesky 分解和式(8)来求得 \mathbf{A}_ω 的解. 首先, 根据 $\mathbf{C}_\omega \mathbf{C}_\omega^T = \mathbf{Q}_\omega \mathbf{Q}_\omega^T$, 将式(8)改写为 $\mathbf{C}_\omega^T \mathbf{C}_\omega \mathbf{A}_\omega = \mathbf{C}_\omega^T \mathbf{F} = \mathbf{z}_\omega$, 通过连续求解两个方程即 $\mathbf{Q}_\omega \mathbf{u} = \mathbf{z}_\omega$ 和 $\mathbf{Q}_\omega^T \mathbf{A}_\omega = \mathbf{u}$ 来更新 \mathbf{A}_ω . 接下来的关键问题是给定 \mathbf{Q}_ω , 需要计算更新 \mathbf{Q}_ω 过程中的 $\mathbf{Q}_{\omega \setminus \{j\}}$ 和 $\mathbf{Q}_{\omega \cup \{i\}}$ 值. 在文献[21]的基础上, 对于 $\mathbf{Q}_{\omega \setminus \{i\}}$ 的计算, 通过 $\mathbf{C}_{\omega \setminus \{i\}} =$

$[\mathbf{C}_w \quad \mathbf{c}_i]$, 可以求得:

$$\begin{aligned} \mathbf{Q}_{w\cup(i)} \mathbf{Q}_{w\cup(i)}^T &= \mathbf{C}_{w\cup(i)}^T \mathbf{C}_{w\cup(i)} \\ &= \begin{bmatrix} \mathbf{C}_w^T \mathbf{C}_w & \mathbf{C}_w^T \mathbf{c}_i \\ \mathbf{c}_i^T \mathbf{C}_w & \mathbf{c}_i^T \mathbf{c}_i \end{bmatrix} = \begin{bmatrix} \mathbf{Q}_w \mathbf{Q}_w^T & \mathbf{C}_w^T \mathbf{c}_i \\ \mathbf{c}_i^T \mathbf{C}_w & 1+\lambda \end{bmatrix} \end{aligned} \quad (15)$$

如果 s 是经由正向替代的 $\mathbf{Q}_w s = \mathbf{C}_w^T \mathbf{c}_i$ 的解, 那么 $\mathbf{Q}_{w\cup(i)}$ 表示为:

$$\mathbf{Q}_{w\cup(i)} = \begin{bmatrix} \mathbf{Q}_w & 0 \\ s^T & \sqrt{(1+\lambda) - s^T s} \end{bmatrix} \quad (16)$$

对于 $\mathbf{Q}_{w\setminus(j)}$ 的计算, 假设:

$$\begin{aligned} \mathbf{C}_w^T \mathbf{C}_w &= \underbrace{\begin{bmatrix} Q_{11} & 0 & 0 \\ l_{21}^T & q_{22} & 0 \\ Q_{31} & l_{32} & Q_{33} \end{bmatrix}}_{\mathbf{Q}_w} \underbrace{\begin{bmatrix} Q_{11}^T & l_{21} & Q_{31}^T \\ 0 & q_{22} & l_{32}^T \\ 0 & 0 & Q_{33}^T \end{bmatrix}}_{\mathbf{Q}_w^T} \\ &= \begin{bmatrix} P_{11} & p_{12} & P_{13} \\ p_{12}^T & k_{22} & p_{32}^T \\ P_{31} & p_{32} & P_{33} \end{bmatrix} \end{aligned} \quad (17)$$

则利用 Cholesky 分解的一阶更新可以快速地由 Q_{33} 和 Q_{32} 得到 \bar{Q}_{33} 。从而得到:

$$\mathbf{Q}_{w\setminus(j)} = \begin{bmatrix} Q_{11} & 0 \\ Q_{31} & \bar{Q}_{33} \end{bmatrix} \quad (18)$$

总之, 通过得到的 w , $\mathbf{Q}_{w\setminus(j)}$ 和 $\mathbf{Q}_{w\cup(i)}$, 很容易求得 \mathbf{Q}_w 。进一步, 根据 $\mathbf{Q}_w \mathbf{u} = \mathbf{z}_w$ 和 $\mathbf{Q}_w^T \mathbf{A}_w = \mathbf{u}$ 来更新 \mathbf{A}_w 。

2.2 自适应字典的训练

近年来, 基于稀疏模型的字典学习算法也是学者们关注的重点, 其中经常被使用的字典学习算法有 DCT 冗余字典、Global 字典和 KSVD 自适应字典。对于 3 种字典算法, 将在实验中进行详细比较。本文采用 KSVD 算法训练字典, 其是一种用来学习图像稀疏表示和去噪的自适应字典算法, 它的本质是通过少量的基向量来表示目标向量。本文利用 KSVD 算法训练字典的步骤可以总结为: 首先, 将大小为 $m \times m$ 的噪声图像分割成大小为 $p \times p$ ($m \gg p$) 的图像块; 其次, 将字典初始化为具有冗余的 DCT 冗余字典, 并利用自适应匹配追踪算法对稀疏系数进行求解; 最后, 根据稀疏系数的解迭代更新字典。

Step1 初始化。设 $\mathbf{X} = \{x_i\}_{i=1}^M$ 为 M 个训练图像块集合, $x \in \mathbf{R}^p$ 表示每个图像块, \mathbf{R}^p 为 p 维向量集。 $\mathbf{A} \in \mathbf{R}^{n \times m}$ 为 \mathbf{X} 的解向量的集合, 且 $\mathbf{A} = \{a_i\}_{i=1}^M$, $a \in \mathbf{R}^n$ 为训练图像块的稀疏系数向量, $\mathbf{B} \in \mathbf{R}^{m \times n}$ 表示字典, 它被初始化 DCT 字典。

Step2 采用自适应匹配追踪算法求解稀疏系数:

$$\begin{aligned} \min \{ \|\mathbf{x}_i - \mathbf{B}\mathbf{a}_i\|_2^2 \} \\ \text{s. t. } \forall i \in w, w \neq \emptyset \end{aligned} \quad (19)$$

其中, w 表示稀疏表示系数非零向量的索引(具体优化见第 2.1.2 节)。

Step3 根据更新的系数, 对字典进行迭代训练。在更新该原子时, 其他原子保持不变, 根据对应的稀疏编码向量, 只更新一个原子。每次迭代更新完字典的所有原子, 就同时更新了系数。设 \mathbf{b}_h 为字典 \mathbf{B} 待更新的第 h 列向量, 即第 h 个原子, 此时式(19)可转化为:

$$\begin{aligned} \|\mathbf{X} - \mathbf{B}\mathbf{A}\|_F^2 &= \|\mathbf{X} - \sum_{i=1}^n \mathbf{b}_i \mathbf{a}_i^T\|_F^2 \\ &= \|\mathbf{X} - \sum_{i \neq h} \mathbf{b}_i \mathbf{a}_i^T - \mathbf{b}_h \mathbf{a}_h^T\|_F^2 \\ &= \|\mathbf{E}_h - \mathbf{b}_h \mathbf{a}_h^T\|_F^2 \end{aligned} \quad (20)$$

其中, \mathbf{a}_h^T 为 \mathbf{b}_h 对应的系数矩阵 \mathbf{A} 中的第 h 行向量, \mathbf{E}_h 是去掉 \mathbf{b}_h 后图像 \mathbf{X} 的分解误差。为了使得字典基向量中非零位置的结构保持不变, 对 \mathbf{a}_h^T 和 \mathbf{E}_h 做去零处理。此时, 需要引入 4 个参数的定义, 如式(21)所示:

$$\begin{aligned} \mathbf{S}_h &= \{i \mid 1 \leq i \leq K, \mathbf{a}_h^T(i) \neq 0\} \\ \mathbf{a}_h^R &= \mathbf{a}_h^T \mathbf{Z}_h, \mathbf{X}_h^R = \mathbf{X} \mathbf{Z}_h, \mathbf{E}_h^R = \mathbf{E}_h \mathbf{Z}_h \end{aligned} \quad (21)$$

设 $\mathbf{S}_h = \{i \mid 1 \leq i \leq K, \mathbf{a}_h^T(i) \neq 0\}$ 为所有图像块集合 $\mathbf{X} = \{x_i\}_{i=1}^M$ 用到原子 \mathbf{b}_h 的索引集, \mathbf{Z}_h 为 $M \times |\mathbf{S}_h|$ 的矩阵, 它在 $(\mathbf{S}_h(i), i)$ 位置上的元素都取 1, 而其他位置的元素取 0。

对 \mathbf{a}_h^T , \mathbf{X} 和 \mathbf{E}_h 做去零处理, 结果分别用 \mathbf{a}_h^R , \mathbf{b}_h^R 和 \mathbf{E}_h^R 表示。 \mathbf{b}_h 的集合表示为 \mathbf{X}_h^R , 去掉样本的误差后表示为 \mathbf{E}_h^R 。因此, 式(21)可以转化为:

$$\|\mathbf{E}_h \mathbf{Z}_h - \mathbf{b}_h \mathbf{a}_h^R \mathbf{Z}_h\|_F^2 = \|\mathbf{E}_h^R - \mathbf{b}_h \mathbf{a}_h^R\|_F^2 \quad (22)$$

对 \mathbf{E}_h^R 做 SVD 分解, 则 $\mathbf{E}_h = \mathbf{U}\mathbf{A}\mathbf{V}^T$, 将 \mathbf{E} 的第一列表示为 $\tilde{\mathbf{b}}_h$, 则 $\tilde{\mathbf{b}}_h$ 是 \mathbf{b}_h 的更新结果。对字典按列依次更新, 直到达到字典更新循环的结束条件, 最终生成新的字典 $\tilde{\mathbf{B}}$ 。

3 基于自适应匹配追踪的去噪模型

通过式(1)可知, 含噪图像可被视为无噪图像与噪声的合成图像。由于含噪图像通常被认为是稀疏的, 即含噪图像可以进行稀疏表示, 而噪声是随机不可稀疏的, 因此利用自适应匹配追踪算法对含噪图像求解稀疏系数, 并利用系数更新自适应字典, 进而重构图像。在重构过程中, 噪声对应的系数被去除, 从而达到去噪的效果。基于自适应匹配追踪去噪模型的伪代码如算法 1 所示。

算法 1 基于自适应匹配追踪的图像去噪算法

输入: 噪声图像 \mathbf{X} , 字典 \mathbf{B}

输出: 去噪图像 $\mathbf{X}_{\text{denoise}}$

Step1 初始化 $w = \{i, \varphi_i < 0\}$, $\mathbf{F} = \begin{bmatrix} \mathbf{X} \\ 0 \end{bmatrix}$, $\mathbf{C} = \begin{bmatrix} \mathbf{B} \\ \sqrt{\lambda \mathbf{I}} \end{bmatrix}$, $\mathbf{Q}_w: \mathbf{C}_w \mathbf{C}_w^T =$

$\mathbf{Q}_w \mathbf{Q}_w^T$ 。

Step2 if $w = \emptyset$, 那么 $\mathbf{Q} = \square$

Step3 While $w \neq \emptyset$ do

更新系数估计 $\mathbf{A}_w: \mathbf{Q}_w \mathbf{u} = \mathbf{z}_w$

$\mathbf{Q}_w^T \mathbf{A}_w = \mathbf{u}$

更新残差: $\mathbf{r}_w = \mathbf{T}_1 (\mathbf{F} - \mathbf{C}_w \mathbf{A}_w) \mathbf{T}_2$

更新字典原子: $\mathbf{E}_h^R = \mathbf{E}_h \mathbf{Z}_h$

$\mathbf{E}_h = \mathbf{U}\mathbf{A}\mathbf{V}^T$

$\mathbf{b}_h = \mathbf{u}_1$

更新索引:

1) 计算

$$[\bar{K}_w, i] \min_{i \in w} \left\{ \varphi_i - \frac{(\mathbf{r}_w^T \mathbf{c}_i)^2}{1+\lambda} \right\}$$

$$[\bar{L}_w, j] = \min_{j \in w} \left\{ (1+\lambda)(a_j^w)^2 + 2c_j^T \mathbf{r}_w a_j^T - \varphi_j \right\}$$

2) 判断索引更新条件

if $\min(\bar{K}_w, \bar{L}_w) \geq 0$, 退出 while 循环

else if $\bar{K}_w < \bar{L}_w$

插入索引: $w = w \cup \{i\}$, 通过式(16)更新 \mathbf{Q}_w

else

删除索引: $w = w \setminus \{j\}$, 通过式(18)更新 \mathbf{Q}_w

end if

end while

Step4 依次更新 $\mathbf{A}_w \rightarrow \mathbf{A}^*$, $\mathbf{B} \rightarrow \mathbf{B}$, $\mathbf{X}^i = \mathbf{B}\mathbf{A}_w \Rightarrow \mathbf{X}_{\text{denoise}} = \tilde{\mathbf{B}}\mathbf{A}^*$ 。

算法1的复杂度主要集中在 Step1 中的 Cholesky 分解和 Step3 中的更新迭代。执行 Cholesky 分解需要的时间复杂度约为 $O(m^2n)$ 。执行 Step3 的每次迭代, 都要完成更新系数估计、更新残差、更新字典原子、更新索引的计算等步骤。在每次迭代中, 更新系数估计的复杂度是常数级, 更新残差中的 \mathbf{T}_1 和 \mathbf{T}_2 可以事先计算出来, 所以迭代的主要复杂度来自于矩阵相乘的计算, 因此时间复杂度约为 $O(m^2n)$ 。更新字典原子的时间复杂度为 $O(msq)$, 其中 s 为字典的原子个数, q 是系数向量中非零元素的个数。更新索引的时间复杂度约为 $O(m^3)$ 。因此, 本文算法的总复杂度可表示为 $O(J(2m^2n + msq + m^3))$, 其中 J 为索引非空迭代次数。

4 实验

为了验证本文算法对图像去噪的性能, 将其与其他方法 (OMP, GMP, RROMP, StOMP, SdOMP, SAMP) 进行对比; 并且采用峰值信噪比 (Peak Signal to Noise Ratio, PSNR)、特征相似性 (Feature Similarity, FSIM)^[28] 和结构相似性 (Structural Similarity, SSIM)^[29] 3 种评价指标对去噪结果进行定量分析; 同时, 通过能量比对算法的分解能力做出比较。实验操作环境是戴尔 Inspiron 3670-R1847s 台式机, 具体配置为: 酷睿八代 i5 处理器, 8 GB 内存, Windows10 操作系统。编程软件为 MATLAB R2017b。

4.1 评价指标

本文采用 3 个客观标准, 即 PSNR, FSIM 和 SSIM 来对去噪结果进行定量评估。

PSNR 是使用最广泛的质量评价标准之一, 它是一种全参考评价方式, 涉及参考图像和待评价图像。给定两个图像 \mathbf{X}_{in} 和 \mathbf{X}_{out} , 它们之间的 PSNR 的定义如下:

$$\text{PSNR} = 10 \log_{10} \frac{255}{\sqrt{\|\mathbf{X}_{\text{in}} - \mathbf{X}_{\text{out}}\|^2}} \quad (23)$$

其中, \mathbf{X}_{in} 表示原始图像, \mathbf{X}_{out} 表示去噪后的图像。PSNR 的值越大, 图像的失真就越小。

FSIM 通过组合相位一致性特征和梯度幅度特征来测量两幅图像之间的相似性。FSIM 的定义如下:

$$\text{FSIM} = \frac{\sum_{x \in \Omega} S_R(x) \times PC_m(x)}{\sum_{x \in \Omega} PC_m(x)} \quad (24)$$

其中, $S_R(x)$ 表示在整个 Ω 空域的相似性, $PC_1(x)$ 和 $PC_2(x)$ 分别表示从 \mathbf{X}_{in} 和 \mathbf{X}_{out} 中提取的相位一致性和梯度幅值。FSIM 取值在 $[0, 1]$ 之间, 值越接近 1, 去噪效果越好。

SSIM 是评价两幅图像的相似程度的指标之一, 它通过亮度、对比度和结构 3 种不同因素对图像的相似度进行评价, 同时利用均值、标准差、协方差分别对亮度、对比度和结构相

似程度进行估计。SSIM 的定义如下:

$$\text{SSIM}(x_{\text{in}}, x_{\text{out}}) = \frac{(2\mu_{x_{\text{in}}}\mu_{x_{\text{out}}} + \theta_1)(2\sigma_{x_{\text{in}}x_{\text{out}}} + \theta_2)}{(\mu_{x_{\text{in}}}^2 + \mu_{x_{\text{out}}}^2 + \theta_1)(\sigma_{x_{\text{in}}}^2 + \sigma_{x_{\text{out}}}^2 + \theta_2)} \quad (25)$$

其中, $\mu_{x_{\text{in}}}$ 是 x_{in} 的平均值, $\mu_{x_{\text{out}}}$ 是 x_{out} 的平均值, $\sigma_{x_{\text{in}}}^2$ 是 x_{in} 的方差, $\sigma_{x_{\text{out}}}^2$ 是 x_{out} 的方差, $\sigma_{x_{\text{in}}x_{\text{out}}}$ 是 x_{in} 和 x_{out} 的协方差, θ_1 和 θ_2 是用于避免分母为 0 造成系统不稳定的常数。SSIM 的取值范围 $[0, 1]$, 同样是值越大, 图像失真越小。

4.2 字典的选择

为了验证自适应字典的优越性, 对大小为 256×256 像素的 Monarch 图片分别加入 15, 25, 35, 50, 70 的高斯噪声, 利用 KSVD 训练大小为 64×256 的字典, 图像块大小为 8×8 , 迭代 50 次。将 DCT 冗余字典、Global 字典与利用 KSVD 算法得到的自适应字典进行比较。图 1 给出了 Monarch 图片的 3 种字典表示, 其中图 1(a) 为干净的图像, 图 1(b) — 图 1(d) 分别为 DCT 冗余字典、Global 字典和利用 KSVD 算法得到的自适应字典。

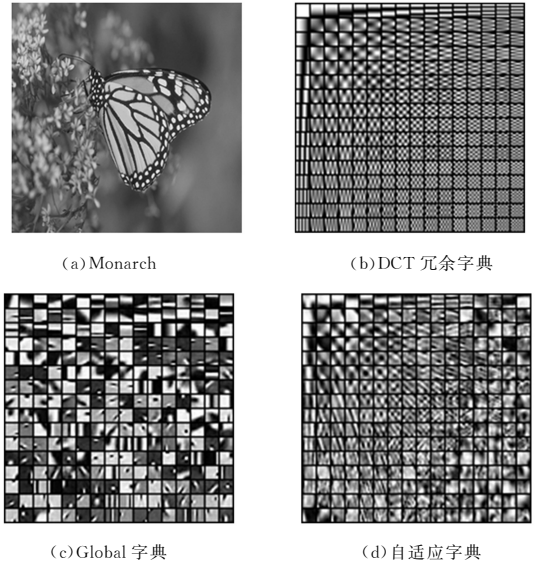


图 1 由 Monarch 图像学习得到的 3 种字典

Fig. 1 Three dictionaries learned from Monarch images

为了比较 3 种字典的去噪性能, 计算在不同的噪声强度下利用 3 种字典去噪的 PSNR 值。根据计算出的结果, 画出对应的折线图, 如图 2 所示。

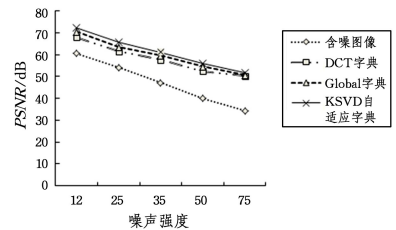


图 2 3 种字典去噪的 PSNR 值的对比

Fig. 2 Comparison of PSNR values for three dictionaries denoising

由图 2 可知, 相比其他两种字典, 利用 KSVD 自适应字典得到的去噪指标更高, 这进一步说明了本文选用 KSVD 自适应字典与自适应匹配追踪算法相结合的有效性。

同时, 本文给出了 Boat 原图像、含有 25 的高斯噪声图像

以及利用本文算法去噪后的图像的灰度直方图,结果如图 3 所示。

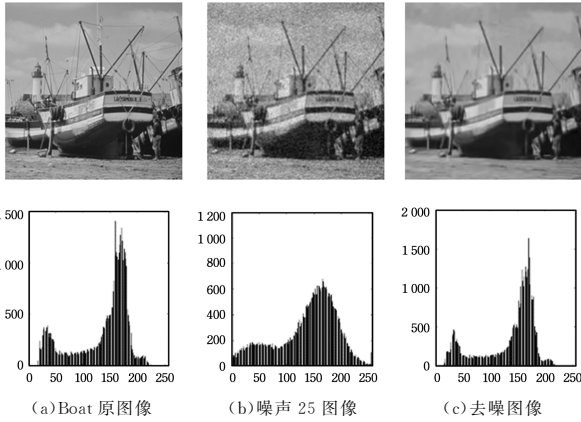


图 3 灰度直方图分布

Fig. 3 Grayscale histogram distribution

图中的结果显示无噪原图像、噪声图像和去噪图像都属于双峰图像,且具有相同的峰顶。可见,本文利用自适应匹配追踪算法重构出的无噪图像与噪声图像拥有相同的主体结构,加噪和去噪之后并不会改变灰度值的分布。这也进一步说明了自适应字典能够高效地利用少量原子来表示整个空间的结构信息。

4.3 实验结果与分析

4.3.1 定性分析

为了充分验证本文算法去噪效果的鲁棒性,通过对 10 张大小为 256×256 像素的图像添加不同噪声强度的高斯白噪声,产生噪声图像。实验所用图像均为灰度图像;图像块设置为 8×8 ;噪声标准差分别为 15, 25, 35, 50, 75。实验结果如图 4 所示。

从图 4 可以看出,在噪声强度都为 25 的条件下,不同算法之间的去噪效果存在差异。

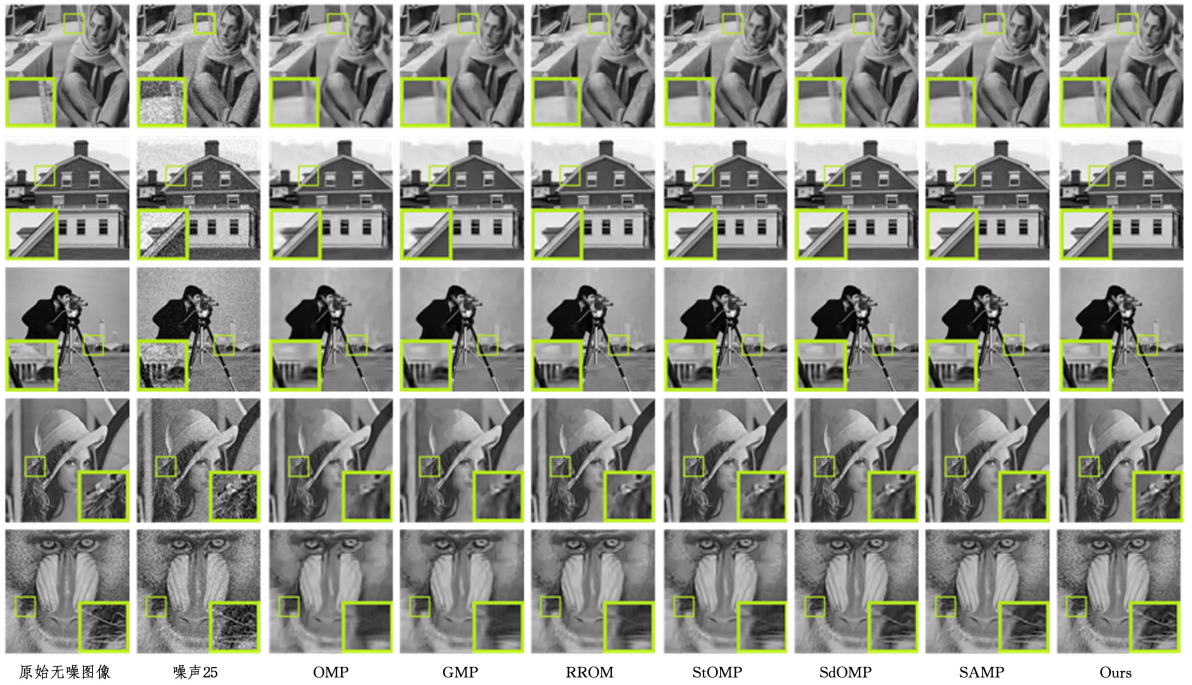


图 4 各种算法对噪声为 25 的 5 种图像的处理结果

Fig. 4 Processing results of five kinds of images with noise of 25 by various algorithms

总体来看,对于多幅不同纹理的图像,本文算法都保持了最佳的视觉效果。OMP 算法处理的图像出现模糊,SAMP 算法与本文算法的处理效果较为接近,同时 SAMP 算法相比其他算法具有竞争优势。

4.3.2 定量评价

为了客观评价去噪图像的质量,本文对图 5 所示的 11 张测试图像进行仿真实验。用 3 种评价指标对去噪结果进行定量比较,结果分别如表 1—表 3 所列。



图 5 测试图片

Fig. 5 Test images

表 1 列出了不同算法的 PSNR 结果。从表中可以看出,在相同的噪声水平下,本文算法较其他算法具有更高的平均 PSNR 值。这也说明利用本文算法处理后,图像的相对失真

程度较小,图像的去噪效果更优。在相同的算法中,随着噪声水平的增大,PSNR 的值逐渐变小,即噪声水平逐渐掩盖了有用的信息,导致图像越来越模糊。表 2 列出了不同算法的

FSIM结果。可以看出,本文提出的算法比其他算法具有更高的FSIM测量;且随着噪声水平的增加,FSIM的值逐渐降低,图像质量变差。表3列出了不同算法的SSIM结果。类似于前两种指标的分析,本文所提算法比其他算法具有更高

的SSIM测量;且噪声水平越高,得到的SSIM值越低,图像质量越差。总之,在视觉效果方面,本文所提算法保留了图像的更多细节,使图像看起来更加自然;从定量的结果来看,其3种指标都具有最高的平均值,与视觉感受一致。

表1 不同方法的PSNR结果

Table 1 PSNR results of different methods

(单位:dB)

Images	Lena	House	Peppers	Boat	Monarch	Woman	Cameraman	Barbara	Bridge	Mandrill	Couple	Average
Noise Level	$\sigma=15$											
OMP	73.967	72.438	56.910	71.074	71.084	78.854	69.592	71.296	63.523	64.032	69.421	69.290
GMP	74.289	72.765	57.295	71.279	71.373	79.285	71.269	72.165	63.931	64.136	69.733	69.774
RROMP	74.665	73.018	57.549	71.438	71.544	79.437	72.165	72.423	64.373	64.289	70.036	70.085
StOMP	75.032	73.565	57.813	72.067	71.871	80.283	72.533	72.891	64.423	64.492	70.346	70.483
SdOMP	75.294	74.009	58.150	72.157	72.064	80.524	72.840	72.925	64.835	64.717	70.864	70.762
SAMP	75.564	74.165	58.212	72.206	72.254	80.864	73.295	73.165	65.246	64.801	71.175	70.995
Ours	75.945	74.298	59.215	72.983	72.460	81.341	75.311	73.962	69.312	64.869	71.252	71.904
Noise Level	$\sigma=25$											
OMP	67.017	65.763	43.496	63.017	62.791	72.194	64.350	64.673	55.793	54.919	61.034	61.368
GMP	68.716	66.946	44.368	63.535	63.139	72.816	64.936	65.247	56.673	55.247	62.266	62.172
RROMP	68.941	66.294	44.946	64.235	63.476	73.846	65.047	65.946	57.948	55.761	62.676	62.647
StOMP	69.047	67.126	45.763	64.843	64.167	74.469	65.584	66.198	58.349	56.942	63.149	63.240
SdOMP	69.346	67.867	46.014	65.135	64.947	74.819	66.729	66.489	58.762	57.347	63.420	63.716
SAMP	69.467	68.406	46.635	65.854	65.432	75.169	67.196	67.944	59.167	58.169	64.765	64.382
Ours	69.630	68.877	47.751	66.039	66.552	76.388	69.574	67.951	60.332	58.473	65.002	65.143
Noise Level	$\sigma=35$											
OMP	61.466	62.109	62.479	54.106	57.020	66.453	59.432	60.729	51.946	51.011	57.468	58.565
GMP	62.776	62.572	63.164	55.905	58.614	68.249	60.316	61.094	52.947	52.661	58.001	58.805
RROMP	62.411	62.956	63.810	56.006	59.456	69.432	61.617	61.379	53.249	52.741	58.334	60.126
StOMP	63.760	63.247	64.572	56.731	60.916	70.761	62.247	62.673	55.190	53.649	59.594	61.213
SdOMP	64.947	63.436	65.349	57.219	61.002	70.964	62.841	63.043	55.672	54.020	60.271	61.706
SAMP	65.350	64.761	65.819	57.915	60.766	71.491	63.761	63.594	56.011	54.864	60.861	62.290
Ours	66.694	65.114	67.212	58.096	61.160	75.649	64.107	64.008	59.222	55.142	61.388	63.436
Noise Level	$\sigma=50$											
OMP	56.014	55.476	54.816	50.167	52.722	62.473	55.302	55.491	49.265	48.365	52.496	53.872
GMP	57.643	56.612	55.276	50.647	53.266	63.612	56.465	56.716	50.724	49.347	54.000	54.937
RROMP	58.164	57.021	55.647	51.763	54.633	64.350	57.016	57.197	51.029	49.846	54.370	55.549
StOMP	59.463	58.246	56.315	52.718	54.976	65.761	58.491	58.146	51.496	50.476	56.210	56.573
SdOMP	60.281	58.647	56.946	53.222	55.147	66.495	58.672	58.469	52.225	51.494	56.493	57.099
SAMP	60.761	59.492	57.817	53.747	55.615	67.015	59.122	59.246	52.679	51.947	56.761	57.655
Ours	61.248	61.182	58.214	54.287	56.212	72.674	60.422	62.901	53.046	52.952	57.030	59.106
Noise Level	$\sigma=75$											
OMP	51.019	50.971	51.372	45.670	46.186	57.494	50.647	49.176	44.399	45.796	47.730	49.133
GMP	52.660	51.764	53.347	46.912	48.210	58.435	52.000	50.090	45.214	46.020	48.291	50.268
RROMP	53.107	52.050	53.863	48.723	48.557	59.643	52.469	51.882	46.942	46.339	49.371	51.177
StOMP	54.972	52.811	55.279	50.162	49.023	62.257	53.844	53.621	47.494	47.646	51.647	52.614
SdOMP	55.497	53.946	55.667	49.006	50.766	62.639	54.729	54.200	49.030	48.555	51.947	53.271
SAMP	56.813	54.015	56.047	49.660	51.433	63.011	54.966	55.224	49.774	49.001	52.146	54.309
Ours	57.308	56.269	58.709	50.202	51.811	69.433	55.313	56.675	50.379	49.798	53.476	55.398

表2 不同方法的FSIM结果

Table 2 FSIM results of different methods

Images	Lena	House	Peppers	Boat	Monarch	Woman	Cameraman	Barbara	Bridge	Mandrill	Couple	Average
Noise Level	$\sigma=15$											
OMP	0.929	0.902	0.820	0.911	0.919	0.927	0.905	0.923	0.902	0.890	0.913	0.904
GMP	0.930	0.911	0.821	0.919	0.928	0.932	0.914	0.933	0.907	0.900	0.914	0.910
RROMP	0.932	0.917	0.822	0.920	0.929	0.937	0.920	0.937	0.914	0.904	0.918	0.914
StOMP	0.936	0.922	0.827	0.925	0.936	0.939	0.922	0.939	0.914	0.905	0.922	0.917
SdOMP	0.942	0.928	0.840	0.927	0.936	0.950	0.925	0.946	0.919	0.909	0.927	0.923
SAMP	0.948	0.936	0.840	0.930	0.941	0.954	0.928	0.946	0.922	0.910	0.929	0.926
Ours	0.950	0.949	0.848	0.954	0.963	0.965	0.950	0.947	0.922	0.943	0.931	0.938

(续表)

Images	Lena	House	Peppers	Boat	Monarch	Woman	Cameraman	Barbara	Bridge	Mandrill	Couple	Average
Noise Level $\sigma=25$												
OMP	0.898	0.877	0.866	0.858	0.864	0.896	0.823	0.877	0.814	0.806	0.782	0.851
GMP	0.901	0.887	0.870	0.868	0.879	0.900	0.867	0.894	0.819	0.812	0.796	0.863
RROMP	0.906	0.900	0.877	0.874	0.882	0.906	0.870	0.901	0.827	0.818	0.800	0.869
StOMP	0.913	0.900	0.881	0.879	0.884	0.918	0.874	0.908	0.831	0.826	0.869	0.880
SdOMP	0.920	0.905	0.889	0.884	0.898	0.922	0.881	0.912	0.837	0.831	0.874	0.887
SAMP	0.922	0.909	0.897	0.890	0.900	0.929	0.886	0.920	0.846	0.834	0.880	0.892
Ours	0.925	0.912	0.924	0.953	0.918	0.955	0.917	0.921	0.865	0.843	0.889	0.911
Noise Level $\sigma=35$												
OMP	0.845	0.847	0.844	0.810	0.829	0.857	0.817	0.843	0.775	0.717	0.786	0.815
GMP	0.853	0.861	0.862	0.820	0.834	0.871	0.830	0.852	0.785	0.739	0.794	0.827
RROMP	0.860	0.869	0.875	0.827	0.849	0.885	0.831	0.863	0.796	0.741	0.807	0.794
StOMP	0.868	0.874	0.879	0.832	0.853	0.891	0.846	0.879	0.807	0.752	0.816	0.845
SdOMP	0.885	0.882	0.884	0.840	0.863	0.897	0.850	0.885	0.811	0.767	0.828	0.854
SAMP	0.890	0.886	0.896	0.859	0.872	0.909	0.860	0.890	0.819	0.773	0.832	0.862
Ours	0.922	0.901	0.904	0.862	0.879	0.948	0.870	0.897	0.821	0.794	0.849	0.877
Noise Level $\sigma=50$												
OMP	0.801	0.825	0.796	0.775	0.772	0.831	0.781	0.803	0.717	0.644	0.761	0.773
GMP	0.812	0.831	0.818	0.782	0.795	0.841	0.788	0.825	0.729	0.650	0.771	0.786
RROMP	0.824	0.837	0.820	0.788	0.819	0.854	0.790	0.833	0.732	0.658	0.776	0.794
StOMP	0.840	0.840	0.831	0.796	0.822	0.869	0.796	0.842	0.742	0.678	0.781	0.803
SdOMP	0.842	0.849	0.837	0.801	0.828	0.876	0.811	0.845	0.749	0.685	0.783	0.810
SAMP	0.853	0.864	0.840	0.806	0.831	0.893	0.826	0.857	0.756	0.692	0.797	0.820
Ours	0.899	0.883	0.861	0.816	0.855	0.950	0.872	0.862	0.766	0.719	0.803	0.844
Noise Level $\sigma=75$												
OMP	0.762	0.755	0.772	0.717	0.715	0.806	0.726	0.759	0.649	0.564	0.669	0.718
GMP	0.771	0.770	0.779	0.729	0.726	0.819	0.730	0.762	0.668	0.582	0.674	0.728
RROMP	0.786	0.786	0.789	0.734	0.732	0.829	0.744	0.772	0.679	0.591	0.695	0.740
StOMP	0.798	0.801	0.795	0.759	0.747	0.849	0.759	0.785	0.685	0.629	0.702	0.755
SdOMP	0.819	0.819	0.811	0.765	0.769	0.855	0.762	0.809	0.695	0.635	0.726	0.770
SAMP	0.824	0.826	0.826	0.771	0.776	0.866	0.788	0.817	0.710	0.627	0.733	0.779
Ours	0.841	0.872	0.855	0.772	0.806	0.895	0.795	0.832	0.729	0.659	0.748	0.800

表3 不同方法的 SSIM 结果

Table 3 SSIM results of different methods

Images	Lena	House	Peppers	Boat	Monarch	Woman	Cameraman	Barbara	Bridge	Mandrill	Couple	Average
Noise Level $\sigma=15$												
OMP	0.681	0.659	0.663	0.640	0.736	0.621	0.680	0.762	0.773	0.744	0.739	0.700
GMP	0.682	0.664	0.663	0.653	0.740	0.631	0.683	0.764	0.779	0.746	0.747	0.705
RROMP	0.688	0.660	0.666	0.659	0.740	0.640	0.688	0.770	0.780	0.749	0.752	0.708
StOMP	0.693	0.669	0.671	0.661	0.742	0.647	0.693	0.774	0.787	0.753	0.756	0.713
SdOMP	0.699	0.670	0.677	0.673	0.748	0.649	0.697	0.782	0.790	0.759	0.760	0.719
SAMP	0.710	0.673	0.680	0.670	0.753	0.650	0.710	0.786	0.796	0.763	0.768	0.724
Ours	0.715	0.708	0.680	0.699	0.759	0.706	0.725	0.790	0.801	0.775	0.770	0.739
Noise Level $\sigma=25$												
OMP	0.607	0.562	0.574	0.546	0.632	0.543	0.601	0.676	0.619	0.571	0.631	0.597
GMP	0.611	0.570	0.573	0.550	0.644	0.546	0.608	0.680	0.623	0.576	0.636	0.602
RROMP	0.617	0.580	0.579	0.554	0.650	0.549	0.610	0.684	0.637	0.577	0.640	0.607
StOMP	0.621	0.586	0.280	0.561	0.657	0.553	0.619	0.699	0.645	0.580	0.647	0.586
SdOMP	0.628	0.593	0.586	0.564	0.666	0.563	0.622	0.704	0.647	0.588	0.650	0.619
SAMP	0.630	0.597	0.597	0.570	0.672	0.570	0.631	0.709	0.652	0.595	0.654	0.625
Ours	0.644	0.610	0.608	0.671	0.668	0.615	0.644	0.711	0.695	0.622	0.705	0.654
Noise Level $\sigma=35$												
OMP	0.534	0.511	0.460	0.447	0.554	0.467	0.533	0.600	0.511	0.433	0.517	0.506
GMP	0.541	0.528	0.476	0.451	0.560	0.474	0.541	0.611	0.529	0.441	0.522	0.516
RROMP	0.543	0.534	0.497	0.463	0.564	0.480	0.548	0.618	0.533	0.452	0.520	0.523
StOMP	0.558	0.541	0.401	0.480	0.576	0.497	0.551	0.621	0.542	0.468	0.528	0.524
SdOMP	0.566	0.543	0.406	0.489	0.591	0.501	0.558	0.628	0.547	0.473	0.536	0.531
SAMP	0.568	0.551	0.409	0.496	0.597	0.518	0.567	0.638	0.551	0.477	0.548	0.538
Ours	0.670	0.574	0.452	0.52	0.605	0.673	0.620	0.740	0.658	0.490	0.653	0.605

(续表)

Images	Lena	House	Peppers	Boat	Monarch	Woman	Cameraman	Barbara	Bridge	Mandrill	Couple	Average
$\sigma=50$												
OMP	0.447	0.451	0.326	0.364	0.474	0.417	0.454	0.501	0.474	0.295	0.422	0.420
GMP	0.451	0.467	0.340	0.369	0.476	0.427	0.458	0.516	0.487	0.304	0.429	0.429
RROMP	0.464	0.461	0.351	0.370	0.477	0.439	0.460	0.530	0.499	0.311	0.431	0.436
StOMP	0.481	0.474	0.368	0.377	0.483	0.440	0.477	0.538	0.514	0.318	0.448	0.447
SdOMP	0.487	0.484	0.371	0.386	0.497	0.452	0.483	0.542	0.520	0.323	0.451	0.454
SAMP	0.490	0.499	0.378	0.397	0.514	0.457	0.488	0.545	0.522	0.327	0.466	0.462
Ours	0.534	0.500	0.480	0.408	0.527	0.491	0.524	0.636	0.576	0.534	0.573	0.528
$\sigma=75$												
OMP	0.331	0.357	0.291	0.275	0.363	0.341	0.340	0.401	0.292	0.188	0.301	0.316
GMP	0.347	0.367	0.300	0.277	0.373	0.351	0.359	0.421	0.303	0.195	0.304	0.327
RROMP	0.360	0.370	0.309	0.286	0.377	0.360	0.361	0.436	0.309	0.197	0.317	0.335
StOMP	0.379	0.388	0.323	0.298	0.392	0.377	0.375	0.445	0.323	0.217	0.328	0.350
SdOMP	0.384	0.396	0.333	0.309	0.406	0.381	0.386	0.454	0.337	0.222	0.342	0.359
SAMP	0.395	0.407	0.346	0.316	0.417	0.398	0.390	0.460	0.349	0.226	0.356	0.369
Ours	0.409	0.414	0.350	0.322	0.422	0.400	0.409	0.469	0.454	0.271	0.361	0.389

图 6—图 8 分别为 OMP, GMP, RROMP, StOMP, SdOMP, SAMP 和本文算法在不同噪声率情况下的 PSNR, FSIM 和 SSIM 平均值变化曲线图。从图中可以看出,随着噪声强度的增加,评价指标的平均值都不断下降。同时,与其他方法相比,本文方法的峰值信噪比平均提高了约 0.76~6.24 dB,特征相似度平均提高了约 0.012~0.082,结构相似性平均提高了约 0.015~0.108。综上,本文方法的性能更优,从而说明它对图像去噪具有明显的优势。

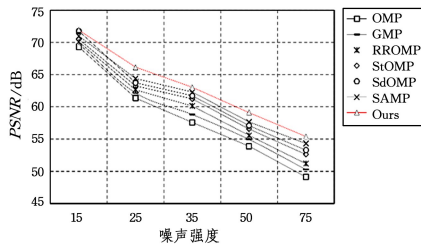


图 6 平均 PSNR 值随噪声强度变化的曲线

Fig. 6 Curve of average PSNR value with noise intensity

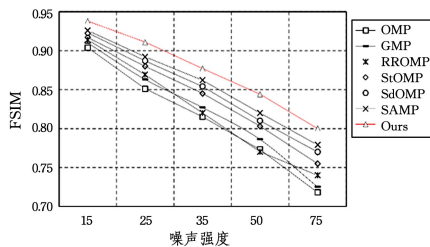


图 7 平均 FSIM 值随噪声强度变化的曲线

Fig. 7 Curve of average FSIM value with noise intensity

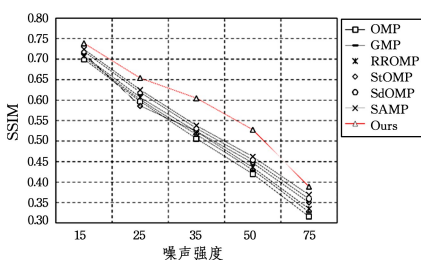


图 8 平均 SSIM 值随噪声强度变化的曲线

Fig. 8 Curve of average SSIM value with noise intensity

4.3.3 迭代分析

为了验证算法的高效性,本文通过能量比与迭代次数的关系来比较各算法的分解效率。在稀疏分解过程中,残差图像的计算过程为 $\mathbf{X}_{\text{res}} = \mathbf{X} - \mathbf{X}_{\text{denoise}}$ (\mathbf{X}_{res} 表示残差图像)。

本文将图像残差与重构的能量比定义为:

$$\rho = \frac{\|\mathbf{X}_{\text{res}}\|^2}{\|\mathbf{X}\|^2} \quad (26)$$

迭代 50 次,观察能量比随迭代次数的变化情况,所得结果如表 4 所列。

表 4 能量比随迭代变化的情况

Table 4 Energy ratio changes with iterations

迭代次数	OMP	GMP	RROMP	SdOMP	StOMP	SAMP	Ours
2	0.357	0.355	0.353	0.344	0.349	0.351	0.342
4	0.329	0.322	0.316	0.314	0.301	0.304	0.281
6	0.264	0.251	0.237	0.224	0.215	0.201	0.159
8	0.193	0.172	0.155	0.146	0.132	0.124	0.050
10	0.172	0.153	0.126	0.114	0.091	0.073	0.031
20	0.127	0.106	0.082	0.073	0.058	0.047	0.025
30	0.106	0.093	0.075	0.063	0.052	0.039	0.021
40	0.067	0.052	0.046	0.038	0.033	0.025	0.019
50	0.063	0.052	0.047	0.038	0.038	0.021	0.015

显然,从表 4 可以看出,与其他几种算法相比,本文算法对能量的分解更加充分,在相同的迭代次数中,分解的效率更高;随着迭代次数的增加,能量比逐渐减小,当迭代达到一定程度时,能量比 ρ 的变化趋于平缓,表明此时图像的主要能量分量被完全分解,通过重建图像将噪声消除,从而得到降噪后的图像。

结束语 本文在深入研究传统图像去噪算法(包括 SAMP, StOMP, SdOMP, RROMP, OMP 和 GMP)的基础上,提出了一种基于先验的贪婪自适应匹配追踪的图像去噪算法。该方法利用自适应匹配算法求解稀疏系数,结合 KSVD 算法训练出自适应字典来重构无噪图像。这种方法能够高效地解决稀疏系数求解问题,且在解决图像去噪问题方面具有优势。实验结果表明,与其他传统方法相比,该算法具有更佳的视觉效果,同时客观评价指标也有所提高。无论是在不同噪声强度影响下,还是在具有不同图像特征的灰度图像下,本文算法都具有较好的去噪效果和更高的去噪效率。

参考文献

[1] JUBAIR I, RAHMAN M, ASHFAQUEUDDIN S, et al. An en-

- hanced decision based adaptive median filtering technique to remove Salt and Pepper noise in digital images[C]//International Conference on Computer & Information Technology. Dhaka: IEEE press,2011:428-433.
- [2] NAVEED K, SHAUKAT B, REHMAN N U. Signal denoising based on dual tree complex wavelet transform and goodness of fit test[C]//International Conference on Digital Signal Processing. London: IEEE,2017:1-5.
- [3] YAGAN A C, OZGEN M T. A spectral graph wiener filter in graph fourier domain for improved image denoising[C]//2016 IEEE Global Conference on Signal and Information Processing (GlobalSIP). Washington: IEEE,2016:450-454.
- [4] DAS S L, NACHIAPPAN A. Role of hybrid switching filter in image denoising - a comparative study[C]//2012 Annual IEEE India Conference (INDICON). Kochi: IEEE,2012:1180-1183.
- [5] CAO Y, LUO Y P, YANG S Y. Hybrid Linear Model Based Image Denoising[J]. Chinese Journal of Computers,2009,32(11):2260-2264.
- [6] RAJA H, BAJWA W U. Cloud K-SVD: A Collaborative Dictionary Learning Algorithm for Big, Distributed Data [J]. IEEE Transactions on Signal Processing,2016,64(1):173-188.
- [7] JIA L N, SONG S T, YAO L H, et al. Image Denoising via Sparse Representation over Grouped Dictionaries with Adaptive Atom Size[J]. IEEE Access,2017,5:22514-22529.
- [8] LU C, SHI J, JIA J. Scale Adaptive Dictionary Learning[J]. IEEE Transactions on Image Processing,2014,23(2):837-847.
- [9] ROMANO Y, ELAD M. Patch-Disagreement as a Way to Improve K-SVD Denoising[C]//IEEE International Conference on Acoustics. Brisbane: IEEE,2015:1280-1284.
- [10] ENGAN K, AASE S O, HUSOY J H. Method of optimal directions for frame design[C]//IEEE International Conference on Acoustics, Speech, and Signal Processing. USA: IEEE,1999:2443-2446.
- [11] AHARON M, ELAD M, BRUCKSTEIN A. MYMrm KMYM-SVD: An Algorithm for Designing Overcomplete Dictionaries for Sparse Representation[J]. IEEE Transactions on Signal Processing,2006,54(11):4311-4322.
- [12] MALLAT S G, ZHANG Z. Matching pursuits with time-frequency dictionaries[J]. IEEE Transactions on Signal Processing,1993,41(12):3397-3415.
- [13] 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.
- [14] DONOHO D L, TSAIG Y, DRORI I, et al. Sparse Solution of Underdetermined Systems of Linear Equations by Stagewise Orthogonal Matching Pursuit[J]. IEEE Transactions on Information Theory,2012,58(2):1094-1121.
- [15] TANG L, CHEN M J. Image Denoising Method Using the Gradient Matching Pursuit [J]. Mathematical Modelling of Engineering Problems,2016,3(2):53-56.
- [16] DENG X, LIU Z. Image denoising based on steepest descent OMP and K-SVD[C]//IEEE International Conference on Signal Processing. Ningbo: IEEE,2015:1-5.
- [17] LI S, FANG L. Signal Denoising With Random Refined Orthogonal Matching Pursuit[J]. IEEE Transactions on Instrumentation and Measurement,2012,61(1):26-34.
- [18] DO T T, GAN L, NGUYEN N, et al. Sparsity adaptive matching pursuit algorithm for practical compressed sensing[C]//Conference on Signals, Systems & Computers. Pacific Grove: IEEE,2008:581-587.
- [19] YUAN S, WANG S, MA M, et al. Sparse Bayesian Learning-Based Time-Variant Deconvolution[J]. IEEE Transactions on Geoscience & Remote Sensing,2017,55(11):6182-6194.
- [20] JIN M, ROTH S, FAVARO P. Noise-Blind Image Deblurring [C]//IEEE Conference on Computer Vision & Pattern Recognition. Honolulu: IEEE,2017:3834-3842.
- [21] VU T H, MOUSAVI H S, MONGA V. Adaptive matching pursuit for sparse signal recovery[C]//IEEE International Conference on Acoustics. New Orleans: IEEE,2017:4331-4335.
- [22] OLIVA G, SETOLA R, HADJICOSTIS C N. Distributed asynchronous Cholesky decomposition[C]//Decision & Control. Las Vegas: IEEE,2016:4414-4419.
- [23] MITCHELL T J, BEAUCHAMP J J. Bayesian Variable Selection in Linear Regression [J]. Publications of the American Statistical Association,1988,83(404):1023-1032.
- [24] GEORGE E, MCCULLOCH R. Variable Selection via Gibbs Sampling[J]. Publications of the American Statistical Association,1993,88(423):881-889.
- [25] CHEN B, PAISLEY J W, CARIN L. Sparse linear regression with beta process priors[C]//IEEE International Conference on Acoustics Speech & Signal Processing. Dallas: IEEE,2010:1234-1237.
- [26] ZHUANG P X, HUANG Y, ZENG D L, et al. Mixed noise removal based on a novel non-parametric Bayesian sparse outlier model[J]. Neurocomputing,2016,174(PB):858-865.
- [27] DING X H, MI Z Y, HUANG Y, et al. Robust rvm based on spike-slab prior [J]. Journal of Electronics (China),2012,29(6):593-597.
- [28] WANG Z, BOVIK A C, SHEIKH H R, et al. Image Quality Assessment: From Error Visibility to Structural Similarity [J]. IEEE Transactions on Image Processing,2004,13(4):600-612.
- [29] ZHANG L, ZHANG L, MOU X Q, et al. FSIM: A Feature Similarity Index for Image Quality Assessment[J]. IEEE Transactions on Image Processing,2011,20(8):2378-2386.



LI Gui-hui, born in 1991, postgraduate, Ph.D supervisor. Her main research interests include graphic image processing and machine learning.



LI Jin-jiang, born in 1978, Ph.D, professor, postgraduate supervisor, is Member of China Computer Federation (CCF). His main research interests include graphic image processing, computer vision and machine learning.