

无监督多重非局部融合的图片去噪方法

陈叶飞¹ 赵广社¹ 李国齐² 王鼎衡¹

摘要 非局部均值去噪 (Non-local means, NLM) 算法利用图像的自相似性, 取得了很好的去噪效果. 然而, NLM 算法对图像中不相似的邻域块分配了过大的权重, 此外算法的搜索窗大小和滤波参数等通常是固定的且无法根据图像内容的变化做出自适应的调整. 针对上述问题, 本文提出一种无监督多重非局部融合 (Unsupervised multi-non-local fusion, UM-NLF) 的图片去噪方法, 即变换搜索窗等组合参数得到多个去噪结果, 并利用 SURE (Stein's unbiased risk estimator) 对这些结果进行无监督的随机线性组合以获得最终结果. 首先, 为了滤除不相似或者相似度较低的邻域块, 本文引入一种基于可微分硬阈值函数的非局部均值 (Non-local means with a differential hard threshold function, NLM-DT) 算法, 并结合快速傅里叶变换 (Fast Fourier transformation, FFT), 初步提升算法的去噪效果和速度; 其次, 针对不同的组合参数, 利用快速 NLM-DT 算法串联生成多个去噪结果; 然后, 采用蒙特卡洛随机采样的思想对上述多个去噪结果进行随机的线性组合, 并利用基于 SURE 特征加权的移动平均滤波算法来抑制多个去噪结果组合引起的抖动噪声; 最后, 利用噪声图像和移动平均滤波后图像的 SURE 进行梯度的反向传递来优化随机线性组合的系数. 在公开数据集上的实验结果表明: UM-NLF 算法去噪结果的峰值信噪比 (Peak signal to noise ratio, PSNR) 超过了 NLM 及其大部分改进算法, 以及在部分图像上超过了 BM3D 算法. 同时, UM-NLF 相比于 BM3D 算法在视觉上产生更少的振铃伪影, 改善了图像的视觉质量.

关键词 图像去噪, 非局部均值, 自相似性, 加权移动平均滤波

引用格式 陈叶飞, 赵广社, 李国齐, 王鼎衡. 无监督多重非局部融合的图片去噪方法. 自动化学报, 2022, 48(1): 87-102

DOI 10.16383/j.aas.c200138

Unsupervised Multi-non-local Fusion Image Denoising Method

CHEN Ye-Fei¹ ZHAO Guang-She¹ LI Guo-Qi² WANG Ding-Heng¹

Abstract Non-local means (NLM) denoising algorithm achieves a good denoising effect with the self-similarity of images. However, NLM assigns excessive weights to dissimilar neighborhood patches in the image. Meanwhile the parameters of NLM, such as the search window size, filtering coefficient and so on, are usually fixed and could not make adaptive adjustments based on changes of image content. In view of the above problems, this paper proposes an unsupervised multi-non-local fusion (UM-NLF) image denoising method, transforming the combined parameters such as the search window to obtain multiple denoising results and making an unsupervised stochastic linear combination for these results by using the Stein's unbiased risk estimator (SURE) to obtain the final result. Firstly, in order to eliminate dissimilar or low similarity neighborhood patches, this paper proposes a non-local means with a differentiable hard threshold function (NLM-DT) algorithm, and then combines fast Fourier transformation to make the preliminary improvements in the denoising effect and speed of the algorithm; Secondly, the paper uses a fast NLM-DT algorithm to generate multiple denoising results in series for different combination parameters; Then this paper combines the above multiple denoising results randomly and linearly with the Monte Carlo random sampling, and uses a weighted moving average filtering algorithm based on SURE features to suppress the jitter noise caused by the combination of multiple denoising results; Finally, the paper uses the SURE of the noise image and the filtered image to optimize linear combination unsupervisedly by the back propagation of gradients. Experiments show that the peak signal to noise ratio (PSNR) of UM-NLF algorithm exceeds those of NLM and most of the improved algorithms of NLM on public datasets and exceeds BM3D on some images. Besides, UM-NLF produces fewer ringing artifacts than BM3D algorithm visually, which improves the visual quality of the image.

Key words Image denoising, non-local means (NLM), self-similarity, weighted moving average filtering

Citation Chen Ye-Fei, Zhao Guang-She, Li Guo-Qi, Wang Ding-Heng. Unsupervised Multi-non-local Fusion Image Denoising Method. *Acta Automatica Sinica*, 2022, 48(1): 87-102

收稿日期 2020-03-16 录用日期 2020-06-11
Manuscript received March 16, 2020; accepted June 11, 2020
国家重点研发计划 (2018TFE0200200), 教育部重大科技创新研究项目, 北京智源人工智能研究院资助
Supported by National Key Research and Development Program of China (2018YFE0200200), Key Scientific Technological Innovation Research Project by Ministry of Education, Beijing Academy of Artificial Intelligence
本文责任编辑 黄庆明

Recommended by Associate Editor HUANG Qing-Ming
1. 西安交通大学电子与信息学部自动化科学与工程学院 西安 710049 2. 清华大学精密仪器系类脑计算研究中心 北京 100084
1. School of Automation Science and Engineering, Faculty of Electronic and Information Engineering, Xi'an Jiaotong University, Xi'an 710049 2. Center for Brain Inspired Computing Research, Department of Precision Instrument, Tsinghua University, Beijing 100084

噪声是正确辨识图像信息的一个障碍,在对图像进行识别、复原或者分割等操作之前,需要对噪声图像进行去噪以提供清晰、准确的图像.目前,已有大量的图像去噪算法^[1-2]被提出,图像的去噪效果和去噪时间也在不断的改善.一般来说,根据具体的去噪算法所在计算域的不同,可以将去噪算法分为变换域方法和空间域方法.

变换域方法是将直观的图像灰度值域信息通过某种方法变换到目标域内,然后对目标域中的信息进行操作,最后再通过逆变换将信息转换回图像本身所在的空间域,如傅里叶变换、小波变换等.虽然变换域方法可以有效去除噪声,保留图像的边缘和纹理信息,但它们也引入了由吉布斯现象^[3]造成的振铃伪影,严重影响了图像的主观视觉质量.然而,空间域方法在抑制噪声的同时,几乎很少产生振铃伪影.近些年, Buades 等^[4]提出的非局部均值的图像去噪算法,充分利用了图像内容的自相似性,通过像素点间相似性权重的加权平均达到去噪的目的.非局部均值算法在去除噪声的同时,较好地保持了原始无失真图像的细节信息,但是 NLM 算法仍存在一些不足:比如算法的时间复杂度过高,导致运行时间过长而无法应用到实际场景;去噪效果过度依赖算法参数的选择等.具体而言,近年来基于非局部均值去噪 (Non-local means, NLM) 算法的改进,主要集中在优化算法的时间复杂度和算法的去噪效果这两个角度.

1) 针对算法时间复杂度的改进,大致可以划分为以下 4 类: a) 减少不相似邻域块权重的计算. Mahmoudi 等^[5]通过计算图像子块的平均灰度和平均梯度进行预分类,剔除不相似的邻域块.虽然平均梯度可以反映图像的细节信息,但是对噪声过于敏感.于是, Coupe 等^[6]采用图像子块的平均灰度和局部方差对邻域进行选择,此方法虽然可以反映像素灰度值的变化,但仍无法表达图像的边缘、纹理等信息.上述两种方法虽然在一定程度上减少了时间复杂度,但是降低了邻域块预选择的有效性,影响了图像去噪的效果.因而, Vignesh 等^[7]研究采用了概率统计的方法滤除不相似的图像子块,改善了邻域估计的可靠性. b) 减少图像数据的维度. Tasdizen^[8]提出的基于主成分分析的非局部均值图像去噪算法,将图像子块投影至低维特征向量空间,并选取一定维数的特征向量计算相似权重系数,在减少计算量的同时提高了去噪效果,缺点在于图像的向量化操作在一定程度上损失了图像空域结构的信息. c) 蒙特卡洛随机采样的思想. Chan 等^[9]利用蒙特卡洛随机采样的思想,根据设定的模型随机选取邻域块的距离,通过计算邻域块距离的小子集来加速算法,对于高维度的图像数据的加速尤为明显;

进一步, Karam 等^[10]利用蒙特卡洛随机卷积核代替固定的值域核,求取多个 NLM 的平均输出,大幅度降低了算法在处理高维度图像时的时间复杂度. d) 此外, Wang 等^[11]利用快速傅里叶变换和积分图来加速图像邻域块间距离的计算,提升了去噪的速度.

2) 针对算法去噪效果的改进,大致可以划分为以下 3 类: a) 采用更合理、鲁棒性更强的邻域块间的相似度量. Chaudhury 等^[12]采用欧氏中值替代欧氏均值作为邻域块距离的度量,克服了 NLM 算法易模糊图像边缘的缺点,尤其适用于噪声较大的情况; Wu 等^[13]提出的 PNLN (Probabilistic non-local means) 采用卡方分布的概率密度函数代替传统 NLM 算法的权重,更好地表示邻域块间的相似性; Deledalle 等^[14]针对不同特征的区域,采用不同几何形状的图像邻域块代替原先的矩形邻域块,在减少时间复杂度的同时提升了去噪的效果; Grewenig 等^[15]利用旋转不变性在块匹配的过程中将候选块旋转一定的角度来与中心邻域块匹配,提升了算法的鲁棒性.除了利用邻域块的几何特性, Wu 等^[16]和 Nguyen 等^[17]分别从不同的角度修正了邻域块中心像素的权重,提升了去噪的效果. b) 算法参数的优化. Salmon^[18]提出一种双参数优化方法,对非局部均值算法的搜索窗大小和中心邻域块的权重进行优化; Van 等^[19]利用斯坦无偏风险估计^[20]作为图像内容的度量来拟合均方差 (Mean square error, MSE) 的变化趋势,从而通过搜索 SURE 最小值时所对应的参数来选择非局部均值算法的滤波参数. c) 结合稀疏表示、低秩分解等理论. Dong 等^[21]联合局部稀疏性,提出一种基于非局部的中心稀疏表示模型,利用图像的自相似性获得原始无失真图像稀疏编码的估计系数,然后使得降质图像的稀疏编码系数不断地逼近这些估计系数,取得了很好的去噪效果; May 等^[22]联合低秩分解理论,提出了一种通过计算非局部均值算子的低阶近似值,提升了去噪效果; Dabov 等^[23]结合频域滤波提出了块匹配三维 (Block-matching and 3D filtering, BM3D) 协同滤波算法,具有较强的纹理细节保持能力,是目前常被用来作为性能对比的基准算法,然而, BM3D 算法在噪声等级较大时,易引入振铃伪影,严重影响去噪图像的视觉质量; Li 等^[24]结合小波分析理论提出了无偏距离的自适应非局部均值算法,进一步提升了去噪的效果.

然而,上述算法并没有解决以下两个问题:

1) 在高斯噪声等级较大时,由于噪声对邻域块计算的影响, NLM 算法对图像中不相似的邻域块分配了过大的权重,容易模糊图像的边缘,极大地影响了图像的去噪效果.

2) 同时,对于不同噪声等级的噪声图像, NLM

算法选取的参数, 比如搜索窗的大小、邻域块的大小和滤波系数等通常是固定且无法根据图像的内容变化做出自适应的调整。

针对上述问题, 本文首先引入基于可微分硬阈值函数的非局部均值算法, 并结合快速傅里叶变换, 初步提升算法的去噪效果和速度; 然后, 提出一种多重可微分硬阈值函数的非局部均值算法的融合方式, 即采用蒙特卡洛随机搜索的思想, 以一定的概率随机变换搜索窗和邻域块等组合参数, 得到多个去噪结果并对其随机线性组合, 而后采用基于 SURE 特征加权的移动平均滤波算法, 抑制组合图像数据的随机浮动; 最后, 利用噪声图像和移动平均滤波后图像的 SURE 值来无监督地优化不同去噪结果的线性组合系数。具体而言, 本文主要有以下三个贡献点:

1) 针对问题 1, 提出了一种可微分的硬阈值函数, 并通过自适应的硬阈值来过滤不相似或者相似度较低的图像邻域块。

2) 针对问题 2, 提出了一种多重可微分硬阈值函数的非局部均值算法的融合方式, 并解决了在未知原始无失真图像下, 利用 SURE 值来无监督、自适应地确定不同去噪图像的线性组合系数的问题。

3) 在公开数据集上, 本文提出的 UM-NLF 算法去噪结果的峰值信噪比 PSNR 超过了 NLM 及其大部分改进算法, 在部分图像上超过了 BM3D 算法。同时, 相比于 BM3D 算法, UM-NLF 算法产生了更少的振铃伪影, 在一定程度上改善了图像的主视觉质量。

本文其余章节的安排如下: 第 1 节主要介绍传统的非局部均值算法的去噪原理。第 2 节介绍无监督多重非局部融合的图像去噪方法的流程, 并详细介绍了流程中各部分的局部细节。第 3 节介绍本文提出的算法在不同高斯噪声等级的噪声图像上的实验结果和分析。第 4 节对本文的方法进行总结, 并提出对未来工作的思考。

1 传统的非局部均值算法

如引言所述, 非局部均值算法主要是去除高斯加性白噪声, 以灰度图像为例, 其噪声图像模型为

$$Y = X + U \quad (1)$$

其中, Y 为含有噪声的图像, X 为原始无失真的图像, U 为服从均值为零, 标准差为 σ 的高斯噪声。

传统的非局部均值去噪算法的具体执行流程如图 1 所示。其中, m 、 n 分别表示灰度噪声图像的高度和宽度, $X(i)$ 、 $X(j)$ 分别表示第 i 个像素点的灰度值和第 j 个像素点的灰度值, Y_i 、 Y_j 分别表示以 i 为中心点的噪声邻域块和以 j 为中心点的噪声邻域块, $w_{i,j}$ 表示第 i 个像素点与第 j 个像素点之间的相似性权重。令 \hat{X} 表示 NLM 的去噪结果, 则对于任

意像素点 $i = (i_1, i_2) \in I$, $\hat{X}(i)$ 为含噪图像 Y 中所有与其具有相似邻域结构的像素点的加权平均, 其相应的数学模型为

$$\hat{X}(i) = \frac{1}{W_i} \sum_{j \in S_i} w_{i,j} \cdot Y(j) = \frac{1}{W_i} \sum_{j \in S_i} \exp\left(-\frac{d(i,j)}{h^2}\right) \cdot Y(j) \quad (2)$$

式中, h 为滤波参数, 其取值与图像的噪声等级和图像的内容有关, 若 h 越大, 则算法的去噪水平越高, 同时也越容易模糊图像的边缘细节; 若 h 越小, 则图像的边缘细节保留的越多, 然而也会残留更多的噪声点, 具体可如图 2 所示。

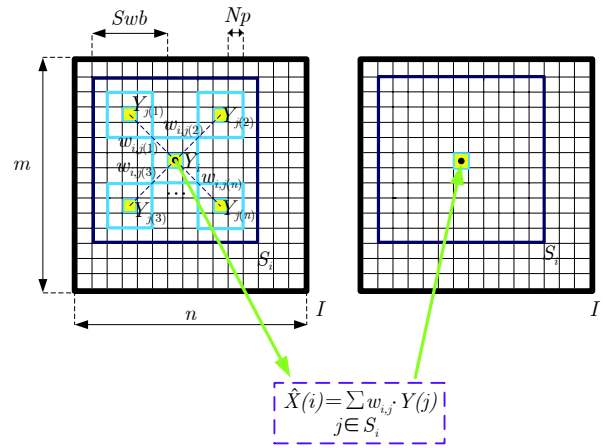


图 1 传统非局部均值算法的执行流程图

Fig.1 Schematic diagrams of non-local means denoising algorithm

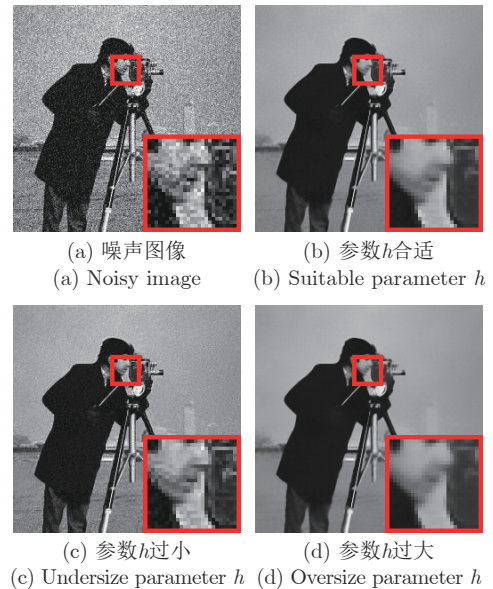


图 2 参数 h 对算法去噪效果的影响

Fig.2 The effect of parameter h on the denoising effect of the algorithm

此外, 在式 (2) 中, S_i 表示以 i 为中心, 半径大小为 Swb 的搜索窗 (Search window block, SWB), W_i 为归一化因子且 $W_i = \sum_{j \in S_i} w_{i,j}$, $d(i, j)$ 表示以 i 为中心的邻域块 Y_i 与以 j 为中心的邻域块 Y_j 的高斯加权欧氏距离平方的度量, 即:

$$d(i, j) = \|Y_i - Y_j\|_{2, \alpha}^2 = \sum_{p=(p_1, p_2) \in P} G_\alpha(p) \cdot (Y(i-p) - Y(j-p))^2 \quad (3)$$

式中, $P = \{(p_1, p_2) | |p_1| \leq Np, |p_2| \leq Np\}$ 是半径大小为 Np 的邻域块 (Neighborhood patch, NP), 且 $G_\alpha(p)$ 为标准差为 α 的高斯核:

$$G_\alpha(p) = \frac{1}{2\pi\alpha^2} \exp\left(-\frac{p_1^2 + p_2^2}{2\alpha^2}\right), p = (p_1, p_2) \in P \quad (4)$$

同时, $\|Y_i - Y_j\|_{2, \alpha}^2$ 的相异性度量^[4] 满足:

$$E \|Y_i - Y_j\|_{2, \alpha}^2 = \|X_i - X_j\|_{2, \alpha}^2 + 2\sigma^2 \quad (5)$$

其中, $E(\cdot)$ 表示数学期望, 式 (5) 保证了这种相异性度量在噪声图像和原始无失真图像概率统计意义上的一致性. 因此, NLM 算法基于非局部的和图像块的去噪策略, 在去除高斯加性白噪声上具有较强的性能.

2 无监督多重非局部融合的去噪方法

NLM 算法对图像中相似度较低的图像邻域块

分配了过大的权重, 容易模糊图像的边缘, 同时 NLM 算法选取的参数, 比如搜索窗的大小、邻域块的大小以及滤波参数等通常是固定的且无法根据图像内容的变化做出自适应的调整. 针对上述问题, 本文提出一种无监督多重非局部算法融合 (UM-NLF) 的图像去噪方法, 即采用蒙特卡洛随机搜索的思想, 以一定的概率随机变换搜索窗等组合参数, 从而得到多个去噪结果并对其随机线性组合, 并利用 SURE 无监督地优化不同去噪结果的组合系数, 具体流程可如图 3 所示. 下文首先介绍了一种基于可微分硬阈值函数的非局部均值算法 (NLM-DT), 并结合快速傅里叶变换, 作为基础模型用来初步提升算法去噪的效果和速度.

2.1 基于可微分硬阈值函数的非局部均值算法

NLM 算法利用高斯加权欧氏距离来度量搜索窗内邻域块间的图像结构相似度. 在噪声等级较低的情况下, 可以减少噪声对邻域块中目标像素点的干扰; 然而由于高斯核函数具有各向同性的特征, 在噪声等级较大时, 算法对原始无失真图像中原本并不相似或者相似度较低的邻域块中目标像素点分配了过大的相似性权重, 往往会模糊图像的边缘和纹理等细节结构部分, 在一定程度上影响了算法去噪的效果.

为了滤除不相似或者相似度较低的图像邻域块, 本文首先引入了一种硬阈值函数 $f_\theta(t), \theta \in [0, 1]$, 对图像邻域块的相似性权重进行硬阈值的截断, 具体的硬阈值函数为

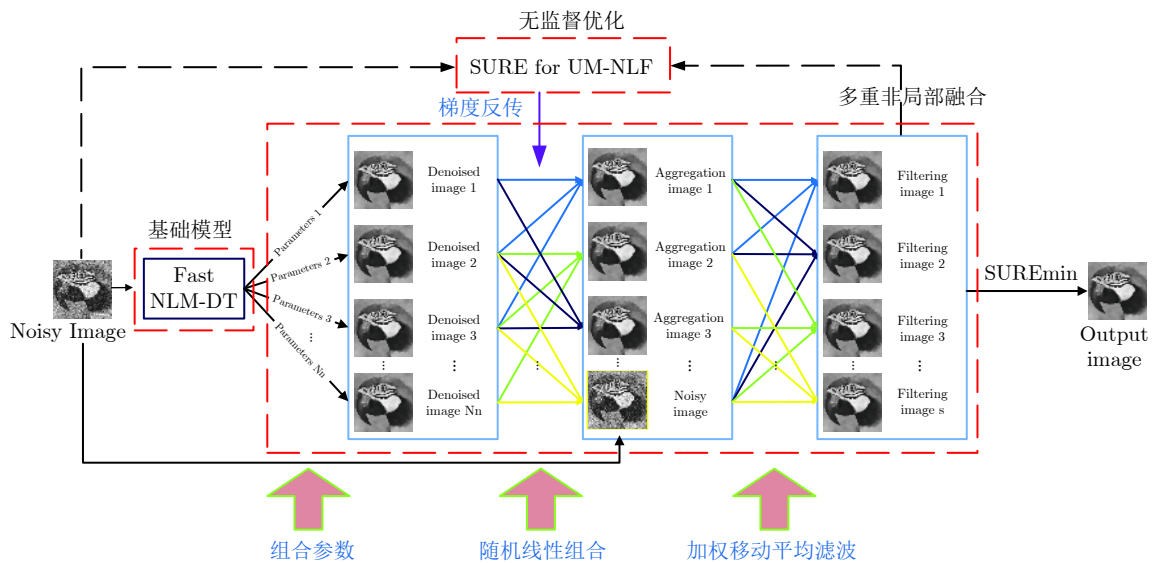


图 3 UM-NLF 去噪算法的整体流程图

Fig. 3 The overall flowchart of the UM-NLF denoising algorithm

$$f_{\theta}(t) = \begin{cases} 1, & \theta < t \leq 1 \\ 0, & 0 \leq t \leq \theta \end{cases} \quad (6)$$

其中, 硬阈值截断前后相似性权重的对比可如图 4 所示.

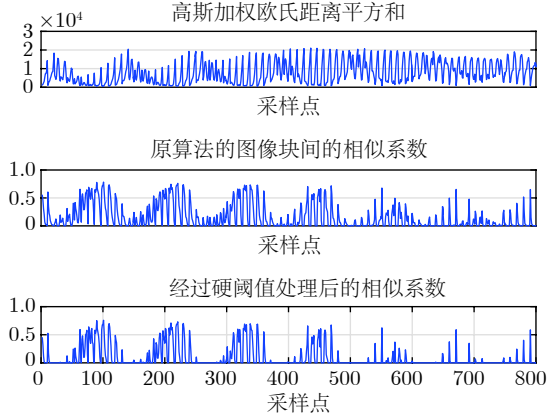


图 4 硬阈值加入前后相似权重值的对比

Fig. 4 Comparisons of similar weight values before and after adding a hard threshold

接着, 将式 (6) 代入式 (2), 化简求得去噪后目标像素点 i 的灰度值 $\hat{X}(i)$:

$$\hat{X}(i) = \frac{\sum_{j \in S_i} w_{i,j} f_{\theta}(w_{i,j}) Y(j)}{\sum_{j \in S_i} w_{i,j} f_{\theta}(w_{i,j})} \quad (7)$$

此时, 为了求解优化后的硬阈值参数 θ , 通常会最小化去噪后的图像与原始无失真图像的各个像素值的均方误差 (Mean square error, MSE), 即:

$$\text{MSE} = \frac{1}{|I|} \sum_{i \in I} (\hat{X}(i) - X(i))^2 \quad (8)$$

式中, $|I|$ 为噪声图像的大小.

在实际情况下, 由于无法预先得到一张含噪图像 Y 的原始无噪声图像 X , 所以无法求解得到去噪图像 \hat{X} 和原始无失真图像 X 的 MSE. 然而, 根据斯坦无偏估计 (SURE) 理论^[20], 其在没有原始无失真图像作为参考的情况下, 仍可以真实反映去噪后图像的质量. 一般而言, 在标准差为 σ 的高斯白噪声条件下, SURE 具体的模型表达式为

$$\text{SURE} = \frac{1}{|I|} \sum_{i \in I} (\hat{X}(i) - Y(i))^2 + \frac{2\sigma^2}{|I|} \sum_{i \in I} \frac{\partial \hat{X}(i)}{\partial Y(i)} - \sigma^2 \quad (9)$$

此时, 如果继续采用上述提出的 $f_{\theta}(t)$, 是无法计算得到其偏导数的, 所以本文构造了一种可微分的硬阈值函数来逼近式 $f_{\theta}(t)$, 优化后的硬阈值函数

$f_{\theta,\lambda}(t)$ 的数学表达式如下:

$$f_{\theta,\lambda}(t) = \frac{1}{1 + e^{-\lambda(t-\theta)}}, \quad 0 \leq t \leq 1 \quad (10)$$

其中, λ 表示可微分硬阈值函数的调节参数. 若 λ 的取值越大, 则 $f_{\theta,\lambda}(t)$ 越接近于 $f_{\theta}(t)$, 即:

$$\lim_{\lambda \rightarrow +\infty} f_{\theta,\lambda}(t) = f_{\theta}(t), \quad 0 \leq t \leq 1 \quad (11)$$

为了充分地验证式 (11) 的相关结论, 本文控制硬阈值参数 θ 不变, 绘制并对比了在不同调节参数 λ 下, 可微分硬阈值函数的分布曲线, 具体可如图 5 所示.

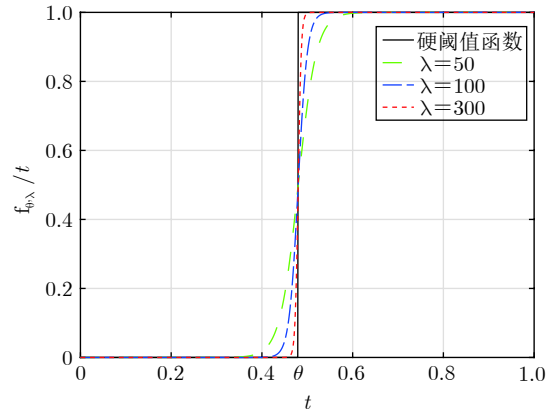


图 5 可微分的硬阈值函数

Fig. 5 Differentiable hard threshold function

可以发现: 若选取的 λ 值越大, 本文所构造的可微分硬阈值函数的分布曲线则越接近于 $f_{\theta}(t)$ 的曲线, 进而验证了式 (11) 的结论.

然后, 本文将可微分的硬阈值函数式 (10) 代入式 (7), 则去噪后目标像素点 i 的灰度值 $\hat{X}(i)$ 可进一步转化为

$$\hat{X}(i) = \frac{\sum_{j \in S_i} w_{i,j} f_{\theta,\lambda}(w_{i,j}) Y(j)}{\sum_{j \in S_i} w_{i,j} f_{\theta,\lambda}(w_{i,j})} \quad (12)$$

基于此, 结合可微分的硬阈值函数, 本文提出了一种基于可微分硬阈值函数的非局部均值算法 (NLM-DT), 来过滤不相似或者相似度较低图像邻域块的权重, 初步提升了算法的去噪效果, 具体的执行流程如算法 1 所示.

算法 1. NLM-DT

输入. 噪声图像 Y .

输出. 去噪图像 \hat{X} .

初始化 $d(i, j) = 0$ 和 $w_{i,j} = 0$.

for 遍历噪声图像的每个目标像素点 i do

根据式 (3), 在以 i 为中心的搜索窗内计算 $d(i, j)$.

计算矩形邻域块 Y_i 与 Y_j 间的相似性权重:

$$w_{i,j} = \exp\left(-\frac{d(i,j)}{h^2}\right)$$

利用式 (12) 依次计算去噪后的像素值 $\hat{X}(i)$.

end for

对于一幅 $m \times n$ 大小的灰度图像, NLM-DT 算法的时间复杂度为 $O(mn(2Swb+1)^2(2Np+1)^2)$, 其中, Swb 表示搜索窗的半径大小, Np 表示邻域块的半径大小. 为了提升 NLM-DT 算法的执行速度, 本文进一步结合文献 [11] 提出的快速傅里叶变换 FFT 来加速高斯核函数和欧氏距离的卷积计算, 减少算法的时间复杂度, 下文简称为 Fast NLM-DT 算法, 具体的执行流程如算法 2 所示, 优化后的时间复杂度为 $O(mn(2Swb+1)^2 \log(mn))$.

算法 2. Fast NLM-DT

输入. 噪声图像 Y .

输出. 去噪图像 \hat{X} .

初始化 $sum_{pixels} = 0$ 和 $sum_{weights} = 0$.

for 所有位移向量 $t_v \in S_i$ **do**

计算位移前后各个像素值的平方差图像:

$$\Delta_{t_v}(i) = (Y(i) - Y(i + t_v))^2, \forall i \in I$$

利用快速傅里叶变换计算 $d(\cdot, \cdot + t_v)$:

$$d(\cdot, \cdot + t_v) = \mathcal{F}^{-1}(\mathcal{F}(G_\alpha(p)) \cdot \mathcal{F}(\Delta_{t_v}(i)))$$

for 所有像素点 $i \in I$ **do**

计算每个像素点间的相似性权重 $w_{i,i+t_v}$:

$$w_{i,i+t_v} = \exp\left(-\frac{d(i, i+t_v)}{h^2}\right)$$

利用可微分硬阈值对相似性权重进行截断:

$$w_{i,i+t_v} \leftarrow w_{i,i+t_v} \cdot f_{\theta,\lambda}(w_{i,i+t_v})$$

更新权重:

$$sum_{weights} \leftarrow sum_{weights} + w_{i,i+t_v}$$

更新像素值总和:

$$sum_{pixels} \leftarrow sum_{pixels} + w_{i,i+t_v} \cdot Y(i + t_v)$$

end for

end for

$$\hat{X}(i) \leftarrow \min(\max(sum_{pixels}/sum_{weights}, 0), 255).$$

2.2 多重可微分硬阈值函数的非局部均值算法融合

上一节提出了一种基于可微分硬阈值函数的非局部均值算法 (NLM-DT), 并结合快速傅里叶变换, 提升 NLM-DT 算法的执行速度. 然而对于不同高斯噪声等级的噪声图像, Fast NLM-DT 算法选取的参数, 比如搜索窗的大小、邻域块的大小以及滤波参数等, 往往是固定的且无法根据噪声图像内容

的变化做出自适应的调整.

文献 [14] 利用主成分分析的方法对图像特征进行降维, 而后对不同形状邻域块去噪后的结果进行局部的融合, 然而此算法的局限性在于对局部邻域块 SURE 精度的评估不足, 从而造成融合权值的不稳定. 因此, 本文在文献 [14] 工作的基础上, 考虑了全局图像的 SURE 估计值, 并借鉴 “Drop Connect”^[25] 的思想: 在神经网络各个网络层节点与节点连接的过程中, 将每个节点与其相连节点的连接权重以一定的概率置零, 故提出了一种多重可微分硬阈值函数的非局部均值算法的融合方式, 即利用蒙特卡洛随机搜索的思想, 以一定的概率随机变换搜索窗、邻域块和滤波参数等组合参数, 从而得到多个去噪结果并对其进行随机线性组合, 具体的执行步骤包括组合参数、随机线性组合和加权移动平均滤波.

2.2.1 组合参数

一般而言, 对于不同高斯噪声等级的噪声图像, 根据不同大小的搜索窗、邻域块、高斯核的系数、滤波参数和自适应的硬阈值参数, 可以获取不同的去噪结果. 因而, 本文不选取某一类参数的全局最优值或局部最优值, 而是选取不同较优参数的组合, 进而利用 Fast NLM-DT 算法串联生成 N_n 个去噪图像, 对应的结果分别是:

$$\hat{X}^{(1)}, \hat{X}^{(2)}, \hat{X}^{(3)}, \dots, \hat{X}^{(r)}$$

同时, 每个去噪图像分别对应一个图像质量的客观评估值 SURE, 以此来估计去噪图像与原始无失真图像的真实 MSE, 即:

$$SURE^{(1)}, SURE^{(2)}, SURE^{(3)}, \dots, SURE^{(r)}$$

其中, $r = 1, 2, 3, \dots, N_n$, 同时 N_n 表示串联生成的去噪结果的数目.

2.2.2 随机线性组合

借鉴 “Drop Connect” 的思想, 本文利用蒙特卡洛随机采样 (Monte Carlo random sampling, MCRS) 的思想, 在一定的概率下随机地选取 N_r ($N_r \leq N_n$) 个不同参数组合的 Fast NLM-DT 算法的去噪结果, 并对它们进行线性组合. 需要注意的是, 为了不丢失原始不失真图像的纹理、边缘等细节特征, 故在每次蒙特卡洛随机组合的过程中, 本文也将噪声图像作为线性组合的一部分, 具体如下:

$$\hat{X}_{mc}^{(q)} = \sum_{z=1}^{N_r} c_z \hat{X}^{(z)} + Y = \sum_{z=0}^{N_r} c_z \hat{X}^{(z)} \quad (13)$$

其中, c_z 表示第 z 个去噪结果的线性组合系数.

然后, 为了充分地提取图像的特征, 本文重复上述线性组合步骤, 进行 N_{mc} 次随机生成, 即可得到如下的去噪结果:

$$\hat{X}_{mc}^{(1)}, \hat{X}_{mc}^{(2)}, \hat{X}_{mc}^{(3)}, \dots, \hat{X}_{mc}^{(q)}$$

其中, $q = 1, 2, 3, \dots, N_{mc}$, 同时, N_{mc} 表示蒙特卡洛随机组合的次数.

2.2.3 加权移动平均滤波

由于蒙特卡洛随机采样具有不确定性, 所以在多幅图像的随机线性组合过程中容易引入额外的抖动噪声. 针对上述问题, 本文提出了一种基于 SURE 特征加权的移动平均滤波 (Weighted moving average filtering, WMAF) 算法, 旨在抑制动态图像组合中随机产生的抖动噪声, 从而保留图像的纹理和边缘等结构细节信息, 尽可能地减少图像数据量的丢失.

对于 N_{mc} 个非平稳图像数据 $\hat{X}_{mc}^{(q)}$, 假设每 N_{wm} 个相邻图像数据在小区间内是接近平稳的, 即其均值接近于常数, 于是可以利用每 N_{wm} 个相邻图像数据的平均值来表示该 N_{wm} 个图像数据中任意一个图像数据. 实际上, 在求解局部图像数据均值的过程中, 每个去噪图像 SURE 值大小也不相同, 比如, 若去噪图像的 SURE 值越大, 则表示此图像的去噪效果不甚理想, 若 SURE 值越小, 则说明此图像的去噪效果越佳.

因此, 根据上述情况, 本文以全长为 N_{mc} 个图像数据为对象, 不断滑动 N_{wm} 个相邻图像数据并做加权平均. 为了便于表示, 令 $L = N_{wm} - 1$, 则移动平滑后的图像为

$$\hat{X}_{wm}^{(s)} = \sum_{l=0}^L \omega^{(s+l)} \hat{X}_{mc}^{(s+l)} \quad (14)$$

其中, $s = 1, 2, \dots, N_{mc} - N_{wm} + 1$, N_{wm} 表示加权移动平均滤波算法选取的基本图像数目单元. 同时, 在式 (14) 中, $\omega^{(s+l)}$ 为权重系数, 表示第 $s+l$ 个图像在 N_{wm} 个图像中所占的比重. 若该图像的 SURE 值越大, 则赋予其的权重系数越小, 即:

$$\omega^{(s+l)} = \frac{1}{\text{SURE}^{(s+l)}} \quad (15)$$

式 (15) 中, $\sum_{l=0}^L \omega^{(s+l)} = 1$.

需要指出的是, 加权移动平均滤波算法的参数选择将直接影响图像的平滑效果, 若移动平滑时所采用的图像数据 N_{wm} 过大, 则可能过度抑制图像随机组合产生的噪声, 图像中的高频变化部分 (比如图像的边缘和纹理细节等) 会被同时弱化, 进而损失图像的局部细节特征; 反之, 若 N_{wm} 取的过小,

则可能对低频的噪声没有进行充分的平滑, 因此应该根据图像数据的实际变化情况, 合理地选取加权移动平均滤波的参数 N_{wm} . 一般而言, 移动平均滤波的参数 N_{wm} 选取的范围在 6 ~ 12 左右, 而在 UM-NLF 算法中取 8 时效果较优.

2.3 无监督优化多重非局部算法的线性组合系数

上一节提出了一种多重可微分硬阈值函数的非局部均值去噪算法的融合方式, 包含组合参数、随机线性组合和加权移动平均滤波这三个部分. 然而并未给出多重非局部算法对应的去噪结果的线性组合系数的优化方案.

因此, 本节在此基础上, 推导并证明了关于 UM-NLF 算法的 SURE 模型表达式, 旨在解决自适应地确定多重非局部算法中不同去噪图像的线性组合系数问题. 接下来, 本节首先解决未知原始无失真图像下对 NLM-DT 算法去噪后图像质量的无偏估计问题, 然后利用上一节加权移动平均滤波后的图像和噪声图像的 SURE 值, 优化多个 Fast NLM-DT 算法对应的不同去噪图像的线性组合系数, 从而达到无监督优化的目的.

2.3.1 NLM-DT 算法的 SURE 模型表达式

为了在未知原始无失真图像的情况下对 NLM-DT 算法去噪后图像的质量进行无偏估计, 本节引入了斯坦无偏估计 (SURE) 理论^[20], 给出了针对 NLM-DT 算法去噪结果的 SURE 模型的一般表达式.

定理 1. 令 $X(i) \in \mathbf{R}$, $i \in I$, $U(i) \sim \mathcal{N}(0, \sigma^2)$, $Y(i) = X(i) + U(i)$, 噪声标准差 σ 已知且 $\hat{X}(i)$ 表示从噪声观测 $Y(i)$ 中预测的估计值, 故可求得估计值 $\hat{X}(i)$ 与实际值 $X(i)$ 的均方误差 MSE 的斯坦无偏估计 SURE 值, 即:

$$\begin{aligned} \text{SURE} &= \frac{1}{|I|} \sum_{i \in I} \left(\hat{X}(i) - Y(i) \right)^2 + \\ &\quad \frac{2\sigma^2}{|I|} \sum_{i \in I} \frac{\partial \hat{X}(i)}{\partial Y(i)} - \sigma^2 \end{aligned} \quad (16)$$

式中, $\hat{X}(i)$ 关于 $Y(i)$ 的偏导数, 即 $\frac{\partial \hat{X}(i)}{\partial Y(i)}$ 为

$$\begin{aligned} &\frac{1}{Q_i} \left(\sum_{j \in S_i, j \neq i} \frac{2w_{i,j} Y(j)}{h^2} \frac{\partial \varphi(w_{i,j})}{\partial w_{i,j}} G_\alpha(0) \cdot (Y(j) - Y(i)) - \right. \\ &\quad \sum_{j \in S_i} \frac{2w_{i,j} Y(j)}{h^2} \frac{\partial \varphi(w_{i,j})}{\partial w_{i,j}} G_\alpha(0) \cdot (Y(j) - Y(i)) + \\ &\quad \left. \sum_{p \in P} \frac{2w_{i,i+p} Y(j)}{h^2} \frac{\partial \varphi(w_{i,i+p})}{\partial w_{i,i+p}} \cdot \right. \\ &\quad \left. (G_\alpha(i-k) \cdot (Y(i-k) - Y(i))) \cdot \right. \\ &\quad \left. (Y(i+k) - \hat{X}(i)) + \varphi(w_{i,i}) \right) \end{aligned} \quad (17)$$

其中, S_i 表示以 i 为中心, 半径大小为 S_{wb} 的搜索窗, $w_{i,j}$ 表示第 i 个像素点与第 j 个像素点之间的相似性权重, $G_\alpha(p)$ 为标准差为 α 的高斯核, 以及 h 为滤波参数. 同时, $Q_i = \sum_{j \in S_i} \varphi(w_{i,j})$, $\varphi(t) = t \cdot f_{\theta,\lambda}(t)$.

证明. 首先, 根据 NLM 算法的基础模型, 如式 (2), 化简求得去噪后目标像素点 i 的灰度值 $\hat{X}(i)$, 即:

$$\hat{X}(i) = \frac{\sum_{j \in S_i} \varphi(w_{i,j}) Y(j)}{\sum_{j \in S_i} \varphi(w_{i,j})} \quad (18)$$

然后, 根据求导法则和链式法则, 对于任意的目标像素点 i , 计算求解 $\hat{X}(i)$ 关于 $Y(i)$ 的偏导数, 化简可得:

$$\begin{aligned} Q_i \frac{\partial \hat{X}(i)}{\partial Y(i)} &= Y(j) \cdot \frac{\partial}{\partial Y(i)} \left(\sum_{j \in S_i, j \neq i} \varphi(w_{i,j}) \right) - \\ &\frac{\partial}{\partial Y(i)} \left(\sum_{j \in S_i} \varphi(w_{i,j}) \right) \cdot \hat{X}(i) + \varphi(w_{i,i}) = \\ &\sum_{j \in S_i, j \neq i} Y(j) \frac{\partial \varphi(w_{i,j})}{\partial Y(i)} + \varphi(w_{i,i}) - \\ &\sum_{j \in S_i} \hat{X}(i) \frac{\partial \varphi(w_{i,j})}{\partial Y(i)} \end{aligned} \quad (19)$$

进一步, 由于

$$Q_i = \sum_{j \in S_i} \varphi(w_{i,j}) \quad (20)$$

所以根据链式法则, 可求得:

$$\frac{\partial \varphi(w_{i,j})}{\partial Y(i)} = \frac{\partial \varphi(w_{i,j})}{\partial w_{i,j}} \cdot \frac{\partial w_{i,j}}{\partial Y(i)} \quad (21)$$

式 (21) 中,

$$\frac{\partial \varphi(w_{i,j})}{\partial w_{i,j}} = \frac{1 + (1 + \lambda)e^{-\lambda(t-\theta)}}{(1 + e^{-\lambda(t-\theta)})^2} \quad (22)$$

其中, λ 表示可微分硬阈值函数的调节参数, θ 表示硬阈值参数. 同时, 针对 $w_{i,j}$ 关于 $Y(i)$ 的偏导数, 可分为以下两种情况进行讨论: 若 $i - j \in P$, 则

$$\begin{aligned} \frac{\partial w_{i,j}}{\partial Y(i)} &= \frac{2w_{i,j}}{h^2} (G_\alpha(0) \cdot (Y(j) - Y(i))) + \\ &\frac{2w_{i,j}}{h^2} (G_\alpha(2i - j) \cdot (Y(2i - j) - Y(i))) \end{aligned} \quad (23)$$

若 $i - j \notin P$, 则

$$\frac{\partial w_{i,j}}{\partial Y(i)} = \frac{2w_{i,j}}{h^2} (G_\alpha(0) \cdot (Y(j) - Y(i))) \quad (24)$$

最后, 本节将 $\varphi(w_{i,j})$ 关于 $w_{i,j}$ 的偏导数, 如式 (22) 的结果表达式, 以及 $w_{i,j}$ 关于 $Y(i)$ 的偏导数, 如式 (23)

和式 (24) 的结果表达式依次代入式 (19), 即可化简得到式 (16) 中的 $\frac{\partial \hat{X}(i)}{\partial Y(i)}$. \square

2.3.2 UM-NLF 算法的 SURE 模型表达式

在求得 NLM-DT 算法去噪结果的 SURE 模型的一般表达式后, 为了自适应地确定多重非局部算法中不同去噪图像的线性组合系数, 本节进一步计算加权移动平均滤波后的图像与噪声图像的 SURE 模型表达式, 具体如下:

$$\begin{aligned} \text{SURE}(\hat{X}_{wm}^{(s)}) &= \frac{1}{|I|} \sum_{i \in I} \left(\hat{X}_{wm}^{(s)}(i) - Y(i) \right)^2 + \\ &\frac{2\sigma^2}{|I|} \sum_{i \in I} \frac{\partial \hat{X}_{wm}^{(s)}(i)}{\partial Y(i)} - \sigma^2 \end{aligned} \quad (25)$$

式中, $\hat{X}_{wm}^{(s)}$ 表示第 s 个经过加权移动平均滤波后的图像, $|I|$ 表示噪声图像的大小, σ 表示噪声图像的噪声标准差.

为了优化 Fast NLM-DT 算法得到的多个去噪结果的线性组合系数, 本节求解 SURE 的最小值, 即利用“梯度下降”的思想将 SURE 值沿着 UM-NLF 算法流程的反方向进行梯度的传递. 相应地, 通过求解式 (25) 关于 c_z 的偏导数, 化简得到如下的表达式:

$$\begin{aligned} \frac{\partial \text{SURE}(\hat{X}_{wm}^{(s)})}{\partial c_z} &= \frac{2}{|I|} \sum_{i \in I} \left(\hat{X}_{wm}^{(s)}(i) - Y(i) \right) \cdot \\ &\left(\sum_{l=0}^L \omega^{(s+l)} \hat{X}^{(z)}(i) \right) + \\ &\frac{2\sigma^2}{|I|} \sum_{i \in I} \frac{\partial}{\partial Y(i)} \left(\sum_{l=0}^L \omega^{(s+l)} \hat{X}^{(z)}(i) \right) \end{aligned} \quad (26)$$

式中, $\omega^{(s+l)}$ 为权重系数, 表示第 $s+l$ 个图像在 N_{wm} 个图像中所占的比重. 同时, N_{wm} 表示加权移动平均滤波算法选取的基本图像数目单元.

由于

$$\sum_{l=0}^L \omega^{(s+l)} = 1 \quad (27)$$

所以对式 (26) 进一步化简, 得到:

$$\begin{aligned} \frac{\partial \text{SURE}(\hat{X}_{wm}^{(s)})}{\partial c_z} &= \frac{2\sigma^2}{|I|} \sum_{i \in I} \frac{\partial}{\partial Y(i)} \left(\hat{X}^{(z)}(i) \right) + \\ &\frac{2}{|I|} \sum_{i \in I} \left(\hat{X}_{wm}^{(s)}(i) - Y(i) \right) \cdot \hat{X}^{(z)}(i) \end{aligned} \quad (28)$$

于是, 推导得出如下的线性系统方程:

$$\sum_{i \in I} \left(\hat{X}_{wm}^{(s)}(i) - Y(i) \right) \hat{X}^{(z)}(i) + \sigma^2 \sum_{i \in I} \frac{\partial \hat{X}^{(z)}(i)}{\partial Y(i)} = 0 \quad (29)$$

其中, $s = 1, 2, \dots, N_{mc} - N_{wm} + 1$, N_{mc} 表示蒙特卡洛随机组合的次数, 以及 N_{wm} 表示加权移动平均滤波算法选取的基本图像数目单元.

综上所述, 本文利用加权移动平均滤波后的图像和噪声图像的 SURE 值, 推导得出与噪声图像、去噪图像和加权移动平均滤波后图像有关的线性系统方程式 (29), 以此来优化多个 Fast NLM-DT 算法产生的不同去噪图像的线性组合系数, 从而达到无监督优化的目的. 具体的, 无监督多重非局部融合 (UM-NLF) 的图像去噪方法的具体执行流程, 如算法 3 所示.

算法 3. UM-NLF

输入. 噪声图像 Y .

输出. 去噪图像 \hat{X} .

步骤 1. 利用算法 2 串联生成 N_n 个去噪结果 $\hat{X}^{(r)}$.

步骤 2. 根据式 (13), 求解得到 $\hat{X}_{mc}^{(q)}$.

步骤 3. 以相同的方式重复步骤 2 进行 N_{mc} 次随机生成.

步骤 4. 根据式 (14), 对 N_{mc} 个去噪结果进行平滑处理.

步骤 5. 优化求解式 (29), 得到最优去噪结果 $\hat{X}_{wm}^{(s)}$.

3 实验结果与分析

本节主要从客观和主观指标出发, 验证提出的 NLM-DT 算法与 UM-NLF 算法的图像去噪性能, 并与原始 NLM^[4] 算法, PNLN^[13]、NLM-SAP^[14]、LJS-NLM^[16]、ANLM^[26] 等具有代表性的非局部均值改进算法, 以及目前传统图像去噪性能最好的非局部算法 BM3D^[23] 进行对比. 同时, 本文也将 UM-NLF 算法与有监督式神经网络图像去噪方法 TNRD^[27] 和 DnCNN-S^[28], 无监督式神经网络图像去噪方法 GCBD^[29] 和 Noise2Void^[30] 进行实验对比和分析. 其中, 每个算法中都包含了自适应的调整操作和大量的参数设置, 在进行对比的过程中, 所有的对比算法均采用原作者提供的原始代码, 同时算法的参数均选取相应论文中的最佳参数.

3.1 实验数据集

为了评估算法对含有高斯加性白噪声灰度图像的去噪性能, 本文以 Classic13 数据集和 BSD68^[31] 数据集为实验对象. 其中, Classic13 数据集是由 13 张被广泛使用的灰度图像数据, 包括 7 张大小为 256×256 的 Cameraman (C.man), House, Peppers, Starfish, Monarch, Airplane 和

Parrot, 以及 6 张大小为 512×512 的 Lena, Barbara, Boat, Man, Couple 和 Baboon. BSD68 数据集包含 68 张不同大小的灰度图像数据. 评价指标主要采用峰值信噪比 PSNR (dB), 结构相似度^[32] (Structural similarity index, SSIM) 和算法的运行时间 (秒).

3.2 实验参数的设置

在实验运行环境的配置方面: 算法的所有对比实验均在一台配置为 Inter(R) Core(TM) i7-6700 CPU 和 8GB RAM 的电脑上运行, 操作系统为 Windows 7 且软件平台为 Matlab 2016b.

在实验算法参数的设置方面: 对于 NLM-DT 算法, 搜索窗的大小选取 21×21 , 邻域块的大小选取 7×7 , 滤波参数选取 $h = \sigma$, 可微分硬阈值函数的调节参数选取 $\lambda = 300$ 以及硬阈值参数选取 $\theta = c \cdot \exp(-\frac{1}{\sigma^2})$, 其中, c 的取值区间为 $[0.02, 0.04]$. 同时, 对于 UM-NLF 算法, 搜索窗大小、邻域块大小、高斯核系数、滤波参数以及可偏导硬阈值函数的参数选择, 具体如表 1 所示. 其中, BSD68 实验数据集的相关参数与表 1 中 512×512 大小图像的处理方式基本一致.

表 1 UM-NLF 算法的参数选择

Table 1 Parameter selection of UM-NLF algorithm

图像大小	参数	参数值
256 × 256	邻域块的直径	[5, 7, 11]
	搜索窗的直径	[13, 21]
	高斯核的系数	[1, 2, 4]
	滤波参数	[0.8, 1.0, 1.2, ..., 2.4]
	硬阈值参数	$[0, 0.02, 0.04] \times \exp(-\frac{1}{\sigma^2})$
	线性组合的数目	40
	蒙特卡洛的次数	80
512 × 512	加权移动平均的数目	8
	邻域块的直径	[5, 7, 11]
	搜索窗的直径	[13, 21]
	高斯核的系数	[1, 2, 4]
	滤波参数	[0.8, 0.95, 1.1, ..., 2.3]
	硬阈值参数	$[0, 0.04, 0.08] \times \exp(-\frac{1}{\sigma^2})$
	线性组合的数目	70
蒙特卡洛的次数	150	
	加权移动平均的数目	8

3.3 传统图像去噪算法的对比

3.3.1 客观指标对比

定量的对比结果具体如表 2 和表 3 所示. 其中, 表 2 和表 3 是不同非局部改进算法在噪声等级 σ 分

别为 10、15、20、25、35 和 50 的灰度噪声图像数据 Classic13 和 BSD68 上去噪效果的对比结果. 从整体而言: UM-NLF 算法几乎超过了所有最新的 NLM 改进算法, 同时在部分噪声图像上, UM-NLF

算法去噪后图像的 PSNR 超过了 BM3D 算法的结果. 从实验的具体细节而言, 主要有三个方面的结论:

1) 算法去噪效果的提升

表 2 在 13 张噪声水平 σ 分别为 10、15、20、25、35 和 50 的噪声图像上不同算法去噪结果的 PSNR (dB)

Table 2 The PSNR (dB) results of different methods on 13 gray images with noise level σ at 10、15、20、25、35 and 50

Level	Images	C.man	House	Peppers	Starfish	Monarch	Airplane	Parrot	Lena	Barbara	Boat	Man	Couple	Baboon
$\sigma = 10$	NLM	31.83	35.35	33.23	32.22	32.47	31.02	31.42	34.85	33.80	32.70	32.66	32.59	28.55
	NLM-SAP	33.50	35.49	34.20	32.44	33.69	32.86	32.92	35.06	33.77	32.97	33.18	33.08	29.78
	PNLM	33.50	35.28	33.66	32.74	33.49	32.85	32.82	34.63	33.39	32.90	33.15	32.78	29.92
	LJS-NLM	33.08	35.18	33.27	32.10	32.54	32.23	32.55	34.54	33.56	32.64	32.82	32.56	30.11
	ANLM	30.33	34.66	32.05	30.69	30.99	29.44	29.84	34.33	32.73	31.88	32.06	31.86	26.97
	BM3D	34.19	36.71	<i>34.68</i>	<i>33.30</i>	<i>34.12</i>	<i>33.33</i>	<i>33.57</i>	35.93	34.98	33.92	33.98	34.04	<i>30.58</i>
	NLM-DT	32.06	35.39	33.23	32.05	33.31	31.03	31.53	34.90	33.63	32.87	32.92	32.74	28.64
	UM-NLF	<i>34.13</i>	<i>36.07</i>	34.71	33.54	34.34	33.54	33.60	<i>35.70</i>	<i>34.44</i>	<i>33.68</i>	33.98	<i>33.64</i>	30.66
$\sigma = 15$	NLM	30.35	33.75	31.35	30.37	30.75	29.59	30.06	32.89	31.67	30.80	30.71	30.49	26.92
	NLM-SAP	31.18	33.85	32.24	30.37	31.28	30.54	30.68	33.32	31.93	31.02	31.03	30.94	27.41
	PNLM	31.25	33.46	31.40	30.42	31.13	30.58	30.64	32.60	31.06	30.86	30.95	30.45	27.59
	LJS-NLM	30.86	33.26	31.05	29.89	30.29	29.93	30.35	32.43	31.19	30.45	30.58	30.11	27.54
	ANLM	29.37	33.34	30.83	29.54	29.79	28.65	28.98	32.87	31.32	30.56	30.58	30.47	26.18
	BM3D	<i>31.91</i>	34.93	<i>32.69</i>	<i>31.14</i>	<i>31.85</i>	<i>31.07</i>	<i>31.37</i>	34.26	33.10	32.13	<i>31.92</i>	32.10	<i>28.18</i>
	NLM-DT	30.55	33.79	31.52	30.34	30.69	29.67	30.12	33.09	31.84	31.08	31.08	30.87	27.18
	UM-NLF	31.92	<i>34.47</i>	32.70	31.36	32.12	31.34	31.44	<i>34.01</i>	<i>32.57</i>	<i>31.82</i>	31.95	<i>31.61</i>	28.32
$\sigma = 20$	NLM	29.24	32.26	29.86	28.68	29.35	28.33	28.96	31.41	29.91	29.32	29.31	28.79	25.44
	NLM-SAP	29.69	32.48	30.77	28.78	29.58	28.94	29.23	31.92	30.38	29.60	29.58	29.31	25.94
	PNLM	29.73	31.95	29.79	28.69	29.51	29.04	29.24	31.15	29.27	29.38	29.44	28.76	26.00
	LJS-NLM	29.35	31.69	29.43	28.15	28.75	28.37	28.92	30.96	29.38	28.91	29.05	28.34	25.82
	ANLM	28.56	32.41	29.65	28.47	28.76	27.86	28.23	31.68	30.12	29.45	29.41	29.25	25.27
	BM3D	30.49	33.77	31.29	<i>29.67</i>	<i>30.35</i>	<i>29.55</i>	<i>29.96</i>	33.05	31.78	30.88	30.59	30.76	<i>26.61</i>
	NLM-DT	29.37	32.51	30.10	28.97	29.38	28.60	29.01	31.68	30.28	29.70	29.74	29.35	25.97
	UM-NLF	<i>30.44</i>	<i>33.29</i>	<i>31.22</i>	29.80	30.61	29.82	30.03	<i>32.77</i>	<i>31.17</i>	<i>30.51</i>	30.59	<i>30.23</i>	26.80
$\sigma = 25$	NLM	28.29	30.89	28.62	27.25	28.19	27.27	28.05	30.26	28.49	28.17	28.24	27.46	24.26
	NLM-SAP	28.61	31.20	29.55	27.43	28.26	27.75	28.24	30.76	29.05	28.47	28.48	27.98	24.74
	PNLM	28.58	30.68	28.52	27.33	28.26	27.83	28.21	30.02	27.85	28.23	28.30	27.46	24.77
	LJS-NLM	28.16	30.39	28.11	26.76	27.53	27.16	27.84	29.86	27.95	27.74	27.91	27.04	24.57
	ANLM	27.91	31.62	28.66	27.49	27.88	27.07	27.57	30.69	29.05	28.50	28.49	28.16	24.38
	BM3D	29.45	32.85	30.16	28.56	<i>29.25</i>	<i>28.42</i>	<i>28.93</i>	32.07	30.71	29.90	29.61	29.71	<i>25.46</i>
	NLM-DT	28.47	31.24	28.95	27.72	28.29	27.70	28.11	30.54	28.97	28.64	28.69	28.10	24.97
	UM-NLF	<i>29.35</i>	<i>32.29</i>	<i>30.04</i>	28.56	29.42	28.67	29.00	<i>31.79</i>	<i>30.07</i>	<i>29.51</i>	<i>29.56</i>	<i>29.12</i>	25.69
$\sigma = 35$	NLM	26.57	28.64	26.66	25.16	26.30	25.48	26.58	28.53	26.35	26.44	26.67	25.67	22.69
	NLM-SAP	26.88	28.88	27.47	25.27	26.30	25.76	26.70	28.94	26.90	26.72	26.89	26.06	22.89
	PNLM	26.76	28.56	26.54	25.32	26.34	25.86	26.65	28.28	25.76	26.47	26.62	25.66	23.02
	LJS-NLM	26.19	28.22	26.11	24.75	25.58	25.16	26.22	28.20	25.86	26.01	26.30	25.30	22.86
	ANLM	26.82	30.05	27.16	25.76	26.43	25.79	26.46	29.16	27.24	26.97	27.11	26.31	22.86
	BM3D	27.92	31.36	28.51	26.86	<i>27.58</i>	<i>26.83</i>	<i>27.40</i>	30.56	28.98	28.43	28.22	28.15	<i>23.82</i>
	NLM-DT	26.98	29.05	27.03	25.65	26.54	26.01	26.73	28.76	26.85	26.91	27.03	26.15	23.38
	UM-NLF	<i>27.77</i>	<i>30.64</i>	<i>28.26</i>	<i>26.70</i>	27.70	26.86	27.54	<i>30.22</i>	<i>28.42</i>	<i>27.96</i>	<i>28.07</i>	<i>27.45</i>	24.09

表 2 在 13 张噪声水平 σ 分别为 10、15、20、25、35 和 50 的噪声图像上不同算法去噪结果的 PSNR (dB) (续表)Table 2 The PSNR (dB) results of different methods on 13 gray images with noise level σ at 10、15、20、25、35 and 50 (continued)

Level	Images	C.man	House	Peppers	Starfish	Monarch	Airplane	Parrot	Lena	Barbara	Boat	Man	Couple	Baboon
$\sigma = 50$	NLM	24.39	26.14	24.48	23.17	24.00	23.36	24.82	26.61	24.26	24.65	25.07	24.06	21.40
	NLM-SAP	24.75	26.32	25.10	23.14	24.07	23.37	24.95	27.07	24.63	24.94	25.36	24.39	21.33
	PNLM	24.67	25.96	24.23	23.17	23.99	23.54	24.84	26.34	23.71	24.61	24.89	23.96	21.48
	LJS-NLM	24.06	25.79	23.84	22.83	23.31	23.06	24.38	26.34	23.83	24.30	24.70	23.71	21.35
	ANLM	25.43	27.95	25.44	23.84	24.82	24.24	25.21	27.61	25.19	25.36	25.69	24.53	21.40
	BM3D	26.13	29.69	26.68	25.04	<i>25.82</i>	25.10	<i>25.90</i>	29.05	27.22	26.78	26.81	26.46	<i>22.35</i>
	NLM-DT	25.08	26.62	24.94	23.69	24.49	24.02	25.04	26.79	24.65	25.05	25.30	24.38	21.81
	UM-NLF	<i>26.06</i>	<i>28.57</i>	<i>26.29</i>	<i>24.80</i>	25.85	<i>24.99</i>	26.04	<i>28.51</i>	<i>26.58</i>	<i>26.31</i>	<i>26.52</i>	<i>25.71</i>	22.53

注: 最佳的两个结果分别以粗体和斜体突出显示

表 3 不同算法在 BSD68 灰度数据集上的平均 PSNR (dB) 和 SSIM

Table 3 Average PSNR (dB) and SSIM results of different methods on BSD68 gray dataset

σ	NLM	NLM-SAP	PNLM	LJS-NLM	ANLM	BM3D	NLM-DT	UM-NLF
10	31.37/0.8809	32.47/0.9006	32.52/0.9000	32.30/0.8984	30.72/0.8834	<i>33.32/0.9163</i>	31.57/0.8934	33.35/0.9184
15	29.64/0.8281	30.29/0.8472	30.25/0.8472	29.95/0.8448	29.44/0.8412	<i>31.07/0.8720</i>	29.96/0.8464	31.16/0.8760
20	28.31/0.7803	28.84/0.7973	28.73/0.7994	28.38/0.7967	28.41/0.8029	<i>29.62/0.8342</i>	28.72/0.8015	29.70/0.8367
25	27.28/0.7368	27.75/0.7550	27.60/0.7563	27.22/0.7539	27.56/0.7685	<i>28.57/0.8017</i>	27.73/0.7585	28.64/0.8034
35	25.75/0.6602	26.17/0.6854	25.95/0.6812	25.58/0.6814	26.25/0.7099	<i>27.08/0.7482</i>	26.20/0.6767	27.12/0.7451
50	24.20/0.5643	24.59/0.6069	24.29/0.5893	23.99/0.5945	24.88/0.6403	<i>25.62/0.6869</i>	24.52/0.5685	25.63/0.6768

由表 2 可知, 在 Classic13 灰度图像数据集上, 当噪声图像的高斯噪声标准差 σ 分别为 10、15、20、25、35 和 50 时, NLM-DT 算法在 13 张灰度噪声图像上去噪后图像的 PSNR (dB) 平均值, 相较于传统非局部均值 NLM 算法的去噪结果, 分别提升了 0.12、0.16、0.29、0.38、0.41 和 0.42; UM-NLF 算法的 PSNR (dB) 平均值, 相比于 NLM 算法, 分别提升了 1.49、1.23、2.67、1.36、1.53 和 1.72。同时, 由表 3 可知, 在 BSD68 灰度图像数据集上, NLM-DT 算法去噪后图像的 PSNR (dB) 平均值, 相较于传统非局部均值 NLM 算法, 分别提升了 0.20、0.32、0.41、0.45、0.45 和 0.32; UM-NLF 算法的 PSNR (dB), 相比于 NLM 算法, 分别提升了 1.98、1.52、1.39、1.36、1.37 和 1.43。这充分验证了本文提出的改进算法 NLM-DT 和 UM-NLF 算法在图像去噪效果上的优越性。

2) 算法去噪的泛化性能

不同去噪算法在 Classic13 和 BSD68 灰度噪声图像集上的实验结果表明: 当噪声图像的高斯噪声标准差较小时, PNLM、NLM-SAP、BM3D 和 UM-NLF 算法去噪结果的峰值信噪比 PSNR 较大, 即取得的图像去噪效果较好; 当噪声图像的高斯噪声标准差较大时, ANLM、BM3D、NLM-DT 和 UM-

NLF 算法呈现出较好的去噪效果。综合而言, BM3D 和 UM-NLF 在不同高斯噪声等级的噪声图像上, 算法去噪的泛化性能最好。

3) 高斯噪声图像的尺寸与去噪效果、去噪时间的关系

对于处在同一个高斯噪声等级的噪声图像, 随着噪声图像尺寸的增大, NLM 算法及其改进算法去噪结果的 PSNR 几乎都有不同程度的提升, 但是算法的去噪时间都有所增加。

3.3.2 视觉效果对比

为了对比算法的视觉效果和纹理细节恢复的能力, 本文选取高斯噪声等级 $\sigma = 20$ 的 Airplane 噪声图像, 高斯噪声等级 $\sigma = 35$ 的 “Test016” 噪声图像 (选自 BSD68 数据集) 和高斯噪声等级 $\sigma = 50$ 的 Baboon 噪声图像。如图 6 所示, 由于噪声图像 Airplane 的噪声等级较低, 并没有对图像的结构造成较大的破坏, 仍保留了原始无失真图像较多的纹理细节, 此时 BM3D 和 UM-NLF 算法都取得了不错的视觉效果; 从图 6 的局部放大图来看, UM-NLF 算法相比 BM3D 算法有着较强的边缘结构保持能力, 比较完整地恢复了 Airplane 图像上飞机型号的字体和飞机尾部的特征。在噪声等级较大时, 如图 7 所示, 由于 “Test016” 噪声图像的背景偏暗, 部分

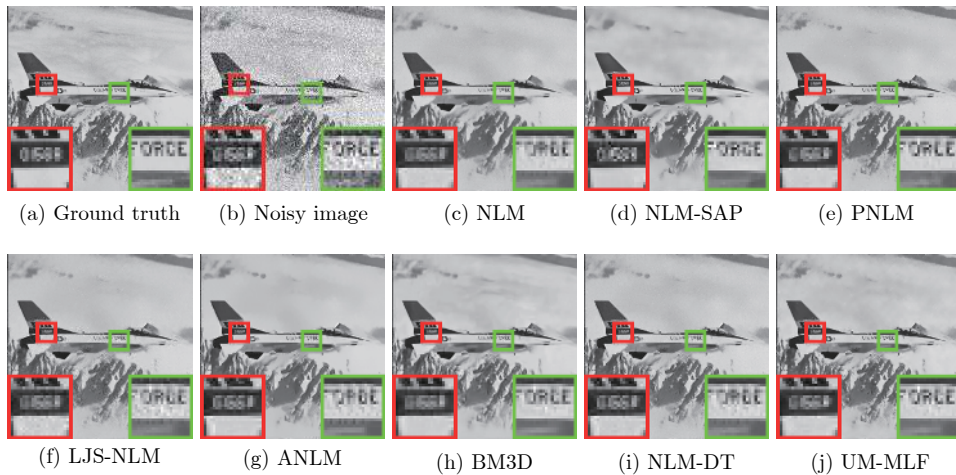


图 6 不同算法对噪声等级 $\sigma = 20$ 的 *Airplane* 噪声图像的去噪结果 PSNR (dB) 和 SSIM ((a) 原始无失真图像; (b) 噪声图像 (22.08 dB/0.4397); (c) NLM (28.33 dB/0.8328); (d) NLM-SAP (28.94 dB/0.8526); (e) PNLM (29.04 dB/0.8490); (f) LJS-NLM (28.37 dB/0.8410); (g) ANLM (27.86 dB/0.8489); (h) BM3D (29.55 dB/0.8755); (i) NLM-DT (28.60 dB/0.8418); (j) UM-NLF (**29.83 dB/0.8760**))

Fig. 6 Denoising results *PSNR* (dB) and *SSIM* on noisy image *Airplane* with noise level $\sigma = 20$ by different methods ((a) Ground truth; (b) Noisy image (22.08 dB/0.4397); (c) NLM (28.33 dB/0.8328); (d) NLM-SAP (28.94 dB/0.8526); (e) PNLM (29.04 dB/0.8490); (f) LJS-NLM (28.37 dB/0.8410); (g) ANLM (27.86 dB/0.8489); (h) BM3D (29.55 dB/0.8755); (i) NLM-DT (28.60 dB/0.8418); (j) UM-NLF (**29.83 dB/0.8760**))

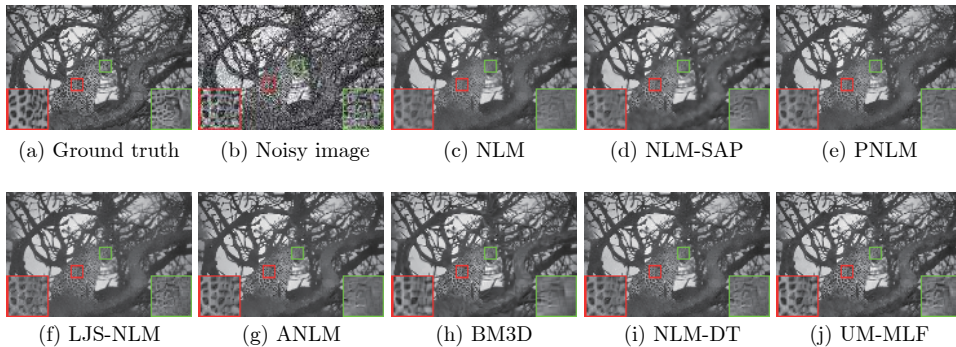


图 7 不同算法对噪声等级 $\sigma = 35$ 的 “*Test016*” 噪声图像的去噪结果 PSNR (dB) 和 SSIM ((a) 原始无失真图像; (b) 噪声图像 (17.25 dB/0.4201); (c) NLM (24.41 dB/0.7202); (d) NLM-SAP (24.75 dB/0.7293); (e) PNLM (24.85 dB/0.7376); (f) LJS-NLM (23.94 dB/0.7168); (g) ANLM (24.89 dB/0.7540); (h) BM3D (25.48 dB/0.7850); (i) NLM-DT (25.04 dB/0.7422); (j) UM-NLF (**25.91 dB/0.7853**))

Fig. 7 Denoising results *PSNR* (dB) and *SSIM* on noisy image “*Test016*” with noise level $\sigma = 35$ by different methods ((a) Ground truth; (b) Noisy image (17.25 dB/0.4201); (c) NLM (24.41 dB/0.7202); (d) NLM-SAP (24.75 dB/0.7293); (e) PNLM (24.85 dB/0.7376); (f) LJS-NLM (23.94 dB/0.7168); (g) ANLM (24.89 dB/0.7540); (h) BM3D (25.48 dB/0.7850); (i) NLM-DT (25.04 dB/0.7422); (j) UM-NLF (**25.91 dB/0.7853**))

结构信息被噪声点所干扰, 由猎豹的脸部和皮肤纹理处的局部放大图可以看出, BM3D 算法存在过平滑的现象, 对图像中的部分结构信息进行了平滑, 其余算法均出现了不同程度的模糊和过度平滑的现象, 而 UM-NLF 算法仍能较好地恢复图像的结构信息. 若噪声等级进一步增大, 如图 8 所示, UM-NLF 算法的优势比较明显, 从 Baboon 图像的局部放大图来看, 狒狒的鼻子和眼睛处恢复的纹理细节更加接近原始无失真图像, 而 BM3D 算法仍有过度平滑

的倾向, 丢失了图像的部分细节信息. 为了进一步验证不同算法的纹理保护能力, 本文采用 XDoG^[33] 滤波对不同算法去噪后的图像进行边缘提取, 具体的结果可见图 9. 可以发现: BM3D 算法处理后的去噪图像与原始无失真图像在经过 XDoG 滤波后的图像, 在狒狒图像的鼻子、眼睛处的边缘纹理有明显的差异.

综合而言, 在视觉效果上, UM-NLF 算法相较于其他改进算法, 具有更强的图像纹理细节的恢复

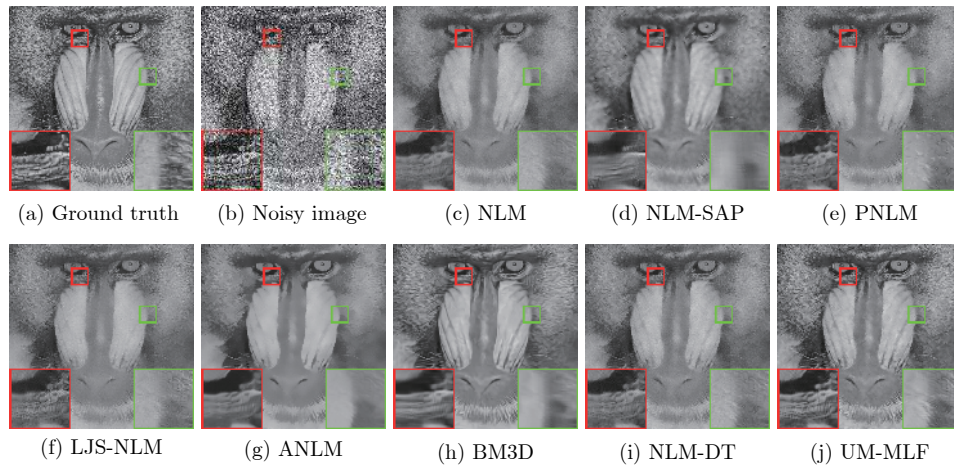


图 8 不同算法对噪声等级 $\sigma = 50$ 的 *Baboon* 噪声图像的去噪结果 PSNR (dB) 和 SSIM ((a) 原始无失真图像; (b) 噪声图像 (14.16 dB/0.2838); (c) NLM (21.40 dB/0.4674); (d) NLM-SAP (21.33 dB/0.4386); (e) PNLM (21.48 dB/0.4793); (f) LJS-NLM (21.35 dB/0.4866); (g) ANLM (21.40 dB/0.4529); (h) BM3D (22.35 dB/0.5489); (i) NLM-DT (21.62 dB/0.4840); (j) UM-NLF (22.53 dB/0.5739))

Fig.8 Denoising results PSNR (dB) and SSIM on noisy image *Baboon* with noise level $\sigma = 50$ by different methods ((a) Ground truth; (b) Noisy image (14.16 dB/0.2838); (c) NLM (21.40 dB/0.4674); (d) NLM-SAP (21.33 dB/0.4386); (e) PNLM (21.48 dB/0.4793); (f) LJS-NLM (21.35 dB/0.4866); (g) ANLM (21.40 dB/0.4529); (h) BM3D (22.35 dB/0.5489); (i) NLM-DT (21.62 dB/0.4840); (j) UM-NLF (22.53 dB/0.5739))

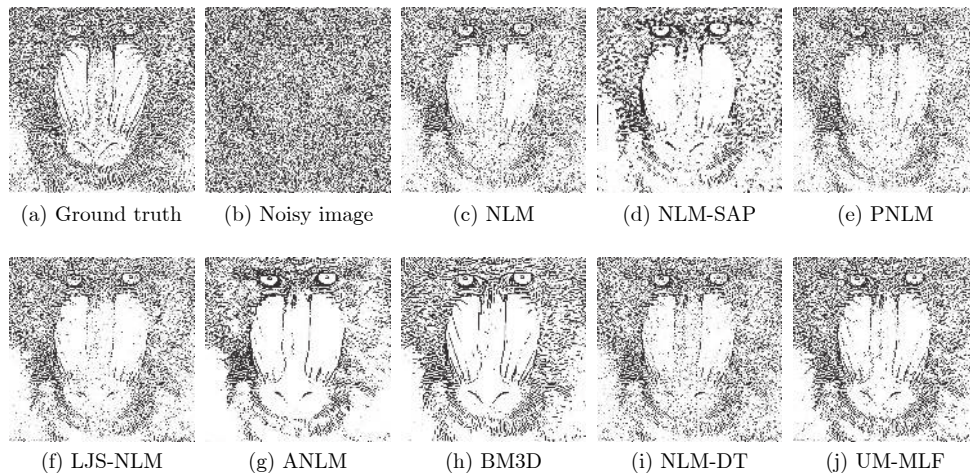


图 9 不同算法对 *Baboon* 噪声图像去噪后并经过 XDoG 滤波的结果

Fig.9 XDoG filtered results of the denoised images of different algorithms on noisy image *Baboon*

能力, 而 BM3D 算法往往会过度平滑图像的边缘细节. 在噪声等级较高时, BM3D 算法由于利用了图像的频域信息, 所以去噪后图像的边缘更可能会产生伪影; 而 UM-NLF 算法相比于 BM3D 算法, 利用了图像的自相似性, 又融合了多个非局部算法去噪图像的空域结构特征, 极大地减少了图像的振铃伪影, 更好地保留了图像的细节.

3.3.3 运行时间对比

算法的运行效率也是评估一个算法的重要因素. 本文比较了 NLM-DT 和 UM-NLF 算法与所有对比算法的运行速度, 具体如表 4 和表 5 所示. 可

以看出: 1) BM3D 算法的去噪速度最快, 对于一张大小为 256×256 的灰度噪声图像, BM3D 算法的处理速度大概在 0.75 秒左右, UM-NLF 算法的去噪时间大概在 25 秒左右. 不过需要注意的是, BM3D 算法利用 C++ 的 MEX 函数以及经过并行处理化的操作, 而 UM-NLF 算法完全使用 Matlab 语言完成. 同时, 由于 UM-NLF 算法根据不同组合参数产生的多个去噪结果, 其相互之间没有硬性的关联, 后续可通过多线程并行处理多个去噪结果或者利用图形处理器 (Graphics processing unit, GPU) 的硬件加速特性, 提升算法的运行速度. 2) 其

表 4 不同算法在大小为 256×256 灰度噪声图像上的去噪时间 (秒)Table 4 Denoising time (seconds) of different algorithms on gray noisy images with a size of 256×256

σ	NLM	NLM-SAP	PNLM	LJS-NLM	ANLM	BM3D	NLM-DT	UM-NLF
10	71.74 ± 0.82	10.89 ± 0.21	7.39 ± 0.64	1.41 ± 0.01	6.35 ± 0.34	0.67 ± 0.09	68.49 ± 1.45	25.55 ± 0.59
15	72.85 ± 3.45	11.11 ± 0.20	7.79 ± 0.58	1.41 ± 0.03	6.31 ± 0.04	0.68 ± 0.09	60.01 ± 1.42	24.41 ± 1.23
20	72.58 ± 4.34	11.23 ± 0.15	8.09 ± 0.46	1.40 ± 0.02	6.25 ± 0.08	0.70 ± 0.09	70.68 ± 0.86	25.27 ± 1.56
25	72.76 ± 1.62	11.36 ± 0.10	8.37 ± 0.58	1.40 ± 0.01	6.26 ± 0.20	0.68 ± 0.11	64.71 ± 2.45	24.98 ± 3.12
35	71.68 ± 0.70	11.43 ± 0.12	8.74 ± 0.31	1.39 ± 0.03	6.25 ± 0.13	0.67 ± 0.10	60.40 ± 0.78	24.68 ± 2.32
50	71.48 ± 1.07	11.48 ± 0.05	8.95 ± 0.40	1.39 ± 0.01	6.27 ± 0.19	0.87 ± 0.05	62.52 ± 1.45	24.30 ± 1.22

表 5 不同算法在大小为 512×512 灰度噪声图像上的去噪时间 (秒)Table 5 Denoising time (seconds) of different algorithms on gray noisy images with a size of 512×512

σ	NLM	NLM-SAP	PNLM	LJS-NLM	ANLM	BM3D	NLM-DT	UM-NLF
10	284.37 ± 13.82	66.46 ± 0.66	38.48 ± 3.46	10.37 ± 0.40	25.27 ± 0.65	3.16 ± 0.13	281.63 ± 11.13	155.64 ± 8.90
15	282.18 ± 11.25	67.77 ± 2.22	41.25 ± 3.55	10.30 ± 0.33	25.46 ± 0.68	3.26 ± 0.11	281.31 ± 10.23	160.28 ± 5.34
20	285.50 ± 11.34	68.70 ± 1.82	42.85 ± 2.50	10.15 ± 0.49	25.70 ± 0.66	3.32 ± 0.11	287.31 ± 13.33	155.94 ± 9.91
25	283.65 ± 12.72	69.53 ± 0.63	44.46 ± 4.01	10.18 ± 0.31	25.54 ± 0.32	3.33 ± 0.08	275.43 ± 14.23	151.56 ± 5.23
35	284.15 ± 15.70	70.41 ± 0.60	46.88 ± 1.89	10.21 ± 0.25	25.40 ± 0.71	3.20 ± 0.17	279.51 ± 13.32	150.65 ± 4.43
50	279.63 ± 19.07	71.46 ± 2.11	49.37 ± 2.83	10.36 ± 0.18	25.40 ± 0.49	3.86 ± 0.09	289.26 ± 14.24	158.75 ± 6.45

余参与对比的非局部均值的改进算法往往以牺牲去噪的精度, 来提升去噪的效率. 3) 同时, 本文发现高斯噪声等级与算法去噪的运行时间并没有一定的关系, 即高斯噪声等级越高, 去噪时间并非越长.

3.4 神经网络图像去噪算法的对比

目前, 以数据驱动的神经网络图像去噪算法^[1-2], 大部分利用配对的噪声图像和原始无失真图像作为训练数据, 通过建立合适的深度神经网络模型 (包含多个卷积层、下采样层、激活函数和全连接层等) 对训练数据进行学习, 而后利用噪声图像和原始无失真图像, 或者噪声图像和残差图像的损失函数对建立的模型进行权重更新, 最后利用训练完毕的模型权重对测试图像进行去噪, 也取得了很好的去噪效果. 接下来, 主要分析 UM-NLF 算法与经典神经网络图像去噪方法执行流程的联系与区别, 并与经典有监督式神经网络图像去噪方法 TNRD^[27] 和 DnCNN-S^[28], 以及无监督式神经网络图像去噪方法 GCBD^[29] 和 Noise2Void^[30] 进行去噪效果的实验对比.

3.4.1 执行流程的对比

本节主要分析和讨论 UM-NLF 算法的执行流程与经典神经网络图像去噪方法执行流程的联系和区别.

1) 联系

从整体的执行结构而言, 本文提出的无监督多重非局部融合 (UM-NLF) 的图像去噪方法, 与经典

神经网络图像去噪方法在执行流程上有类似的结构. 具体而言, 基于可微分硬阈值函数的非局部均值算法, 通过度量以待处理像素点为中心的图像邻域块与搜索区域内图像邻域块的相似性, 而对具有相似邻域结构的像素点进行加权平均, 这一过程可类比于神经网络模型的“卷积层”模块, 后者通过不同的卷积核提取图像的特征并过滤噪声; 此外, 随机线性组合可类比于“全连接层”和“Drop Connect”模块, 后者用于对提取的特征进行组合并输出滤波结果; 同时, 本文提出的基于 SURE 加权的移动平均滤波算法, 还可类比于“下采样层”模块, 后者用于减少算法的参数并抑制组合图像数据的随机浮动. 这种结构上的相似性, 在一定程度上反映了 UM-NLF 算法具有与神经网络类似的特征提取能力.

2) 区别

本节主要从算法的执行方式和计算资源的角度, 对比 UM-NLF 算法与经典神经网络图像去噪方法的区别. a) 执行方式 UM-NLF 算法充分考虑图像空域结构信息的自相似性, 利用多重非局部算法的随机线性组合, 并利用 SURE 自适应地调节不同去噪图像的线性组合系数, 无需训练, 直接对测试图像进行去噪, 具有很好的鲁棒性能; 而经典的神经网络图像去噪方法往往需要在大量的数据样本上训练, 通过建立相应的深度神经网络模型, 包括卷积层、下采样层和激活函数等基础模块的拼接, 并利用反向传播算法学习噪声图像至原始无失真图像 (或者残差图像) 的函数映射. b) 计算资源 经典

表 6 传统算法和深度学习算法在 BSD68 灰度数据集上的平均 PSNR (dB) 和 SSIM

Table 6 Average PSNR (dB) and SSIM results of traditional and deep learning methods on BSD68 gray dataset

σ	传统的无监督去噪方法		无监督式神经网络去噪方法		有监督式神经网络去噪方法	
	BM3D	UM-NLF	Noise2Void	G CBD	TNRD	DnCNN-S
15	31.07/0.8720	31.16/0.8760	-/-	31.59/-	31.42/0.8822	31.73/0.8906
25	28.57/0.8017	28.64/0.8034	27.71/-	29.15/-	28.92/0.8148	29.23/0.8278
50	25.62/0.6869	25.62/0.6762	-/-	-/-	25.97/0.7021	26.23/0.7189

注: “-”表示原始论文中没有给出相应的实验结果

的神经网络图像去噪方法往往需要在大量的训练数据上进行学习, 主要利用训练数据的先验信息, 往往依赖高算力的计算资源, 以及特定的训练数据集; 而 UM-NLF 算法在普通的 CPU 上即可运行, 不过分依赖计算资源, 同时 UM-NLF 算法对噪声图像具有很好的鲁棒性.

3.4.2 去噪效果的对比

在 BSD68 的灰度图像数据集上, 本文将提出的 UM-NLF 算法与传统的无监督图像去噪算法 BM3D, 有监督式神经网络图像去噪方法 TNRD 和 DnCNN-S, 以及无监督式神经网络去噪方法 G CBD 和 Noise2Void 进行去噪效果对比和实验分析. 具体的去噪结果如表 6 所示, 可以发现: 目前而言, 有监督式神经网络图像去噪方法取得了较好的去噪效果. 然而, 传统的无监督算法 BM3D 和 UM-NLF 在缺少特定场景训练样本的情况下, 在实际应用中仍然可以成为灵活而有希望的选择.

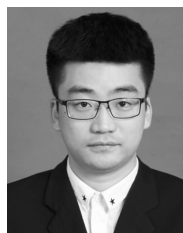
4 总结

针对原始 NLM 算法对图像中相似度较低的邻域块分配了过大的权重, 以及传统算法固定的参数无法根据图像内容的变化进行自适应调整的问题, 本文提出了一种基于 SURE 参数优化的无监督多重非局部融合 (UM-NLF) 的图像去噪方法, 即利用蒙特卡洛随机搜索的思想, 变换搜索窗等组合参数得到多个去噪结果并利用 SURE 来无监督地优化多重非局部算法去噪结果的组合系数. 实验结果表明: 在公开数据集上, 相比于目前传统算法去噪效果最佳的 BM3D 算法, 本文提出的 UM-NLF 算法产生了更少的振铃伪影, 有效保持了图像的纹理和结构信息, 极大地改善了图像的主观视觉质量. 目前, 由于提出的算法主要集中在灰度图像的讨论, 未来的工作主要是考虑研究如何将提出的算法拓展至彩色图像的去噪, 比如研究如何结合彩色图像 R、G、B 通道的相关性, 以及采用何种的距离判别准则来衡量两个彩色图像子块的相似性权重.

References

- 1 Tian C W, Fei L K, Zheng W X, Xu Y, Zuo W M, Lin C W. Deep learning on image denoising: an overview [online], available: <http://arxiv.org/abs/1912.13171>, January 16, 2020
- 2 Dong W S, Wang P Y, Yin W T, Shi G M, Wu F F, Lu X T. Denoising prior driven deep neural network for image restoration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2019, **41**(10): 2305–2318
- 3 Geman S, Geman D. Stochastic relaxation, gibbs distributions, and the Bayesian restoration of images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1984, **6**(5-6): 721–741
- 4 Buades A, Coll B, Morel J M. A non-local algorithm for image denoising. In: Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. San Diego, California, USA: IEEE, 2005.60–65
- 5 Mahmoudi M, Sapiro G. Fast image and video denoising via non-local means of similar neighborhoods. *IEEE Signal Processing Letters*, 2005, **12**(12): 839–842
- 6 Coupe P, Yger P, Prima S, Hellier P, Kervrann C, Barillot C. An optimized blockwise nonlocal means denoising filter for 3-D magnetic resonance images. *IEEE Transactions on Medical Imaging*, 2008, **27**(4): 425–441
- 7 Vignesh R, Oh B T, Kuo C C J. Fast non-local means (NLM) computation with probabilistic early termination. *IEEE Signal Processing Letters*, 2010, **17**(3): 277–280
- 8 Tasdizen T. Principal neighborhood dictionaries for nonlocal means image denoising. *IEEE Transactions on Image Processing*, 2009, **18**(12): 2649–2660
- 9 Chan S H, Zickler T, Lu Y M. Monte Carlo non-local means: random sampling for large-scale image filtering. *IEEE Transactions on Image Processing*, 2014, **23**(8): 3711–3725
- 10 Karam C, Hirakawa K. Monte-Carlo acceleration of bilateral filter and non-local means. *IEEE Transactions on Image Processing*, 2018, **27**(3): 1462–1474
- 11 Wang J, Guo Y W, Ying Y T, Liu Y L, Peng Q S. Fast nonlocal algorithm for image denoising. In: Proceedings of the 2006 IEEE International Conference on Image Processing. Atlanta, GA, USA: IEEE, 2006.1429–1432
- 12 Chaudhury K N, Singer A. Non-local euclidean medians. *IEEE Signal Processing Letters*, 2012, **19**(11): 745–748
- 13 Wu Y, Tracey B, Natarajan P, Noonan J P. Probabilistic non-local means. *IEEE Signal Processing Letters*, 2013, **20**(8): 763–766
- 14 Deledalle C A, Duval V, Salmon J. Non-local methods with shape-adaptive patches (NLM-SAP). *Journal of Mathematical Imaging and Vision*, 2012, **43**(2): 103–120
- 15 Grewenig S, Zimmer S, Weickert J. Rotationally invariant similarity measures for nonlocal image denoising. *Journal of Visual Communication and Image Representation*, 2011, **22**(2): 117–130
- 16 Wu Y, Tracey B, Natarajan P, Noonan J P. James-stein type

- center pixel weights for non-local means image denoising. *IEEE Signal Processing Letters*, 2013, **20**(4): 411–414
- 17 Nguyen M P, Chun S Y. Bounded self-weights estimation method for non-local means image denoising using minimax estimators. *IEEE Transactions on Image Processing*, 2017, **26**(4): 1637–1649
- 18 Salmon J. On two parameters for denoising with nonlocal means. *IEEE Signal Processing Letters*, 2010, **17**(3): 269–272
- 19 Van D V D, Kocher M. SURE-based non-local means. *IEEE Signal Processing Letters*, 2009, **16**(11): 973–976
- 20 Stein C M. Estimation of the mean of a multivariate normal distribution. *The Annals of Statistics*, 1981, **9**(6): 1135–1151
- 21 Dong W S, Zhang L, Shi G M, Li X. Nonlocally centralized sparse representation for image restoration. *IEEE Transactions on Image Processing*, 2013, **22**(4): 1620–1630
- 22 May V, Keller Y, Sharon N, Shkolnisky Y. An algorithm for improving non-local means operators via low-rank approximation. *IEEE Transactions on Image Processing*, 2016, **25**(3): 1340–1353
- 23 Dabov K, Foi A, Katkovnik V, Egiazarian K. Image denoising by sparse 3-D transform-domain collaborative filtering. *IEEE Transactions on Image Processing*, 2007, **16**(8): 2080–2095
- 24 Li X Y, Zhou Y C, Zhang J, Wang L H. Multipatch unbiased distance non-local adaptive means with wavelet shrinkage. *IEEE Transactions on Image Processing*, 2020, **29**: 157–169
- 25 Wan L, Zeiler M D, Zhang S, Yann L C, Fergus R. Regularization of neural networks using dropconnect. In: Proceedings of the 2013 IEEE International Conference on Machine Learning. Atlanta, GA, USA: IEEE, 2013.1058–1066
- 26 Xing Xiao-Xiao, Wang Hai-Long, Li Jian, Zhang Xuan-De. Asymptotic non-local means image denoising algorithm. *Acta Automatica Sinica*, 2020, **46**(9): 1952–1960 (邢笑笑, 王海龙, 李健, 张选德. 渐近非局部平均图像去噪算法. 自动化学报, 2020, **46**(9): 1952–1960)
- 27 Chen Y J, Pock T. Trainable nonlinear reaction diffusion: A flexible framework for fast and effective image restoration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2017, **39**(6): 1256–1272
- 28 Zhang K, Zuo W M, Chen Y J, Meng D Y, Zhang L. Beyond a Gaussian denoiser: Residual learning of deep CNN for image denoising. *IEEE Transactions on Image Processing*, 2017, **26**(7): 3142–3155
- 29 Chen J W, Chen J W, Cao H Y, Yang M. Image blind denoising with generative adversarial network based noise modeling. In: Proceedings of the 2018 IEEE Conference on Computer Vision and Pattern Recognition. Salt Lake City, Utah, USA: IEEE, 2018. 3155–3164
- 30 Krull A, Buchholz T O, Jug F. Noise2Void-learning denoising from single noisy images. In: Proceedings of the 2019 IEEE Conference on Computer Vision and Pattern Recognition. Long Beach, CA, USA: IEEE, 2019. 2129–2137
- 31 Roth S, Black M J. Fields of experts. *International Journal of Computer Vision*, 2009, **82**(2): 205–229
- 32 Wang Z, Bovik A C, Sheikh H R, Simoncelli E P. Image quality assessment: From error visibility to structural similarity. *IEEE Transactions on Image Processing*, 2004, **13**(4): 600–612
- 33 Winnemoller H, Kyprianidis J E, Olsen S C. XDoG: An extended difference-of-Gaussians compendium including advanced image stylization. *Computers & Graphics*, 2012, **36**(6): 740–753



陈叶飞 西安交通大学电子与信息学部自动化科学与工程学院硕士研究生. 主要研究方向为模式识别, 图像去噪.

E-mail: yefeichen@stu.xjtu.edu.cn

(**CHEN Ye-Fei** Master student at the School of Automation Science and Engineering, Faculty of Electronic and Information Engineering, Xi'an Jiaotong University. His research interest covers pattern recognition, image denoising.)



赵广社 西安交通大学电子与信息学部自动化科学与工程学院教授. 主要研究方向为深度学习与图像识别, 大数据与群体智能, 多智能体系统协同导航与控制.

E-mail: zhaogs@mail.xjtu.edu.cn

(**ZHAO Guang-She** Professor at the School of Automation Science and Engineering, Faculty of Electronic and Information Engineering, Xi'an Jiaotong University. His research interest covers deep learning, image recognition, big data, swarm intelligence, multi-agent system cooperative navigation and control.)



李国齐 清华大学精密仪器系类脑计算研究中心副教授. 主要研究方向为信号处理, 类脑计算, 复杂网络. 本文通信作者.

E-mail: liguoqi@mail.tsinghua.edu.cn

(**LI Guo-Qi** Associate professor at the Center for Brain Inspired Computing Research, Department of Precision Instrument, Tsinghua University. His research interest covers signal processing, brain-like computing, and complex network. Corresponding author of this paper.)



王鼎衡 西安交通大学电子与信息学部自动化科学与工程学院博士研究生. 主要研究方向为图像处理, 模型优化.

E-mail: wangdai11@stu.xjtu.edu.cn

(**WANG Ding-Heng** Ph. D. candidate at the School of Automation Science and Engineering, Faculty of Electronic and Information Engineering, Xi'an Jiaotong University. His research interest covers image processing and model optimization.)