

# 基于相异性阈值的改进自适应稀疏表示去噪算法

吴雄洲, 李跃华\*

(南京理工大学 电子工程与光电技术学院, 江苏 南京 210094)

**摘要:** 针对自适应稀疏表示去噪算法在对图像进行去噪时运行时间较长, 得到结果过于平滑的问题, 研究了基于相异性阈值的改进自适应稀疏表示去噪算法, 在改进算法中, 计算当前提取的图像块与前一个图像块之间的相异性度量, 并与阈值进行比较, 低于阈值则认为两者具有相同的稀疏表示向量和表示误差, 不需要对当前块再执行计算从而减少运行时间, 高于阈值则认为当前块包含了边缘区域, 记录其位置, 在重构去噪图像时予以保护, 以减少图像边缘信息的损失. 对毫米波图像的去噪实验结果证实了改进算法的有效性.

**关键词:** 相异性阈值; 改进; 自适应稀疏表示; 去噪; 运行时间; 边缘保护

**中图分类号:** TP391 **文献标识码:** A

## Modified adaptive sparse representation denoising algorithm based on difference threshold

WU Xiong-Zhou, LI Yue-Hua\*

(Dept. of Electronic and Optical Engineering, Nanjing University of Science and Technology, Nanjing 210094, China)

**Abstract:** In order to decrease the time and alleviate the smoothness of the adaptive sparse representation algorithm, a modified adaptive sparse representation algorithm based on difference threshold is introduced in this paper. This modified algorithm computes the difference between the current block and the previous one, then compares the difference with the threshold. When the difference is less than the threshold, the two blocks are considered having the same sparse representation vector and error. It is not needed to compute over current block again. When the difference is greater, they are considered as different. The current block contains the edge area and its position is recorded. It is then protected from averaging in reconstructing the result to alleviate the smoothness. The experimental results performed on millimeter-wave image demonstrated the effectiveness of the proposed method.

**Key words:** difference threshold, modified, adaptive sparse representation, denoising, operation time, edge protection

**PACS:** 07.05.Pj

## 引言

在稀疏理论提出以前, 对图像进行去噪处理主要是采用各种滤波技术, 如维纳滤波, 双边滤波, 中值滤波等等, 图像在经过这些滤波算法处理后会产生一定程度的失真, 导致去噪效果不理想. 当稀疏理论出现并成功应用到图像去噪领域以后, 降低了这种失真, 并且相比于以往的滤波算法能更大程度的

抑制噪声, 改善去噪效果, 因此逐渐成为了图像处理领域的研究热点. Michal Aharon, Michael Elad 和 Alfred Bruckstein 提出了 K-SVD 算法<sup>[1]</sup>来设计过完备字典, Michael Elad 和 Michal Aharon 在此基础上提出了以噪声图像为对象来训练自适应字典, 然后在该字典上对含噪图像进行稀疏表示去噪<sup>[2]</sup>的算法, 并取得了较好的效果. 但是由于该算法在采用“滑窗”的方式对含噪图像进行分块后并没有判断图像

收稿日期: 2015-11-16, 修回日期: 2016-04-19

基金项目: 总装备部预研基金(51305050102)

Foundation items: Supported by Pre-Research Foundation of General Armament Department, China (51305050102)

作者简介 (Biography): 吴雄洲 (1991-), 男, 陕西渭南人, 硕士, 主要研究领域为数字图像处理. E-mail: wuxiongzhou123@sina.com

\* 通讯作者 (Corresponding author): E-mail: hmb4507@mail.njust.edu.cn

Received date: 2015-11-16, revised date: 2016-04-19

块彼此是否近似相同,使得对最终结果有着近似相同贡献的图像块都执行了计算,导致算法运行时间较长,此外,该算法还采取了对每个去噪图像块先进行取平均操作再“拼接”成最终去噪结果的处理方式,导致去噪图像的边缘信息受损,平滑性上升.

这里研究了基于相异性阈值的改进自适应稀疏表示去噪算法,与上述自适应稀疏表示去噪算法相比,加入了与相异性阈值进行判断的步骤,目的是为了区分不同和近似相同的图像块以便执行不同的处理操作,从而在缩短算法运行时间的同时也保护图像的边缘信息,减轻去噪结果的平滑性.

## 1 基于相异性阈值的改进自适应稀疏表示去噪算法

### 1.1 相异性度量 $\beta_d$ 和相异性阈值 $th_d$

为了给后续阐述改进的自适应稀疏表示去噪算法做准备,这里引入图像块相异性度量和相异性阈值的概念.相异性度量,指的是相互重叠的两个图像块彼此的相异程度,记为  $\beta_d$ .相异性阈值,指的是相互重叠的两个图像块区分彼此同异的分界线,记为  $th_d$  ( $th_d \in [0, 1]$ ).设  $x_{ij}$  是从图像  $X$  的位置  $(i, j)$  处提取出来的图像块(大小为  $\sqrt{n} \times \sqrt{n}$ ),  $x_{i'j'}$  是紧挨着  $x_{ij}$  以“滑窗”方式从图像  $X$  的位置  $(i', j')$  处提取出来的图像块(大小为  $\sqrt{n} \times \sqrt{n}$ ),  $x_{ij}$  和  $x_{i'j'}$  彼此相互重叠,则两者之间的相异性度量可以表示为:

$$\beta_d = \frac{\|\text{reshape}(x_{i'j'}, n, 1) - \text{reshape}(x_{ij}, n, 1)\|_0}{n}, \quad (1)$$

其中,记  $\eta = \text{reshape}(x_{i'j'}, n, 1) - \text{reshape}(x_{ij}, n, 1)$ ,  $\text{reshape}(\cdot)$  是 matlab 中的函数,其功能是在保持元素数目不变的前提下重排矩阵的形状;  $\|\cdot\|_0$  表示向量中非零元素的数目,  $\beta_d \in [0, 1]$  为两重叠图像块之间的相异性度量,即两个图像块中对对应位置具有不同灰度值的像素点在图像块中所占的比例,  $n$  为每个图像块所含的像素点数目.当  $\beta_d \geq th_d$  时,认为两个图像块不同,即一个相对另外一个有明显变化(包含了图像的边缘区域),当  $\beta_d < th_d$  时,则认为两个图像块近似相同,对于同一个自适应字典有相同的稀疏表示向量和误差.

### 1.2 算法实现过程

本文的基于相异性阈值的改进自适应稀疏表示去噪算法(简记为改进算法, Modified Algorithm)作为对自适应稀疏表示去噪算法(简记为自适应算

法, Adaptive Algorithm)的改进,不同之处在于在算法中引入了相异性阈值判断的步骤,改进算法具体描述如下:

噪声模型:  $Y = X + V$ , 其中  $Y$  为含噪图像,  $X$  为原始图像,  $V$  为加性高斯白噪声.

算法参数:  $n$ -图像块尺寸,  $k$ -字典尺寸,  $th_d$ -相异性阈值, 取  $th_d = 0.6$ ,  $J$ -训练迭代次数, 取  $J = 10$ ,  $\sigma$ -噪声标准差,  $\lambda$ -拉格朗日乘子, 取  $\lambda = 30/\sigma$ ,  $C$  为噪声增益, 取  $C = 1.15$ .

求解的目标函数:

$$\min_{\alpha} \left\{ \lambda \|Y - X\|_2^2 + \sum_{ij} \mu_{ij} \|\alpha_{ij}\|_0 + \sum_{ij} \|D\alpha_{ij} - R_{ij}X\|_2^2 \right\} \quad (2)$$

(1) 初始化: 设  $Y = X$ ,  $D =$  过完备离散余弦变换(DCT)字典, 相异性阈值  $th_d = 0.6$ ,  $\sigma$  由预先设定值或者估计值确定, 设当前提取出来的图像块为  $R_{ij}X$ , 其稀疏表示向量为  $\alpha_{ij}$ , 表示误差为  $e_{ij}^s$ , 前一个提取出来的图像块为  $R_{i'j'}X$ , 稀疏表示向量为  $\alpha_{i'j'}$ , 表示误差为  $e_{i'j'}^s$ .

(2) 重复  $J$  次:

① 稀疏编码阶段: 使用正交匹配追踪算法(OMP)求解式(6)得到图像块的稀疏表示向量:

While( $(i, j)$  还未到达最后一个图像块位置)

$$\left\{ \begin{array}{l} \text{计算 } R_{ij}X \text{ 和 } R_{i'j'}X \text{ 之间的相异性度量 } \beta_d: \\ \left\{ \begin{array}{l} \eta = \text{reshape}(R_{ij}X, n, 1) - \text{reshape}(R_{i'j'}X, n, 1) \\ \beta_d = \frac{\|\eta\|_0}{n} \end{array} \right. \\ \text{if}(\beta_d < th_d) \left\{ \right. \end{array} \right. \quad (3)$$

当前块和前一块的稀疏表示向量相同:

$$\alpha_{ij} = \alpha_{i'j'} \quad (4)$$

$\left. \begin{array}{l} \left. \right\} \\ \text{else} \left\{ \right. \end{array} \right\}$

将当前图像块的位置记录到  $\delta$  集合中:

$$\delta = \{(i, j) | \beta_d \geq th_d\} \quad (5)$$

计算当前块的稀疏表示向量:

$$\alpha_{ij} = \underset{\alpha_{ij}}{\text{V}} \min \|\alpha_{ij}\|_0 \text{ subject to } \|R_{ij}X - D\alpha_{ij}\|_2^2 \leq (C\sigma)^2 \quad (6)$$

$\left. \begin{array}{l} \left. \right\} \\ \left. \right\} \end{array} \right\}$

② 字典更新阶段: 对于字典  $D$  中的每一列

$s = 1, 2, 3 \dots k$ , 通过以下方式来更新:

I. 找到使用这个字典原子(字典的每一列称为一个原子)的图像块集合:

$$\omega_s = \{(i, j) | \alpha_{ij}(s) \neq 0\} \quad (7)$$

II. 对于每个满足  $(i, j) \in \omega_s$  的索引, 计算表示误差:

$$\begin{aligned} & \text{While}((i, j) \in \omega_s) \{ \\ & \quad \text{if}(\beta_d < th_d) \{ \\ & \quad \quad \text{当前块和前一块的稀疏表示误差相同:} \\ & \quad \quad \quad e_{ij}^s = e_{i'j'}^s, \quad (8) \end{aligned}$$

$$\begin{aligned} & \quad \quad \quad \} \\ & \quad \text{else} \{ \\ & \quad \quad \text{计算当前块的稀疏表示误差:} \end{aligned}$$

$$e_{ij}^s = R_{ij}X - \sum_{m \neq s} d_m \alpha_{ij}(m) \quad (9)$$

III. 将式(9) 计算出来的表示误差以列向量的形式构成误差矩阵  $E_s$ , 即  $E_s$  的列是  $\{e_{ij}^s\}_{(i,j) \in \omega_s}$ .

IV. 对误差矩阵进行奇异值分解:  $E_s = U\Delta V^T$ , 选取  $U$  的第一列作为更新后的字典原子  $\tilde{d}_s$ , 取  $V$  的列乘以  $\Delta(1, 1)$  后来更新系数值  $\{\alpha_{ij}(s)\}_{(i,j) \in \omega_s}$ .

(3) 最终恢复的去噪图像为:

$$\begin{aligned} X = & (\lambda I + \sum_{ij} R_{ij}^T R_{ij})^{-1} (\lambda Y + \sum_{ij} R_{ij}^T D \alpha_{ij})_{(ij) \notin \delta} + \\ & (\sum_{ij} D \alpha_{ij})_{(ij) \in \delta} \quad (10) \end{aligned}$$

上述改进算法中, 嵌入了 C 语言的描述方式, 但这只是一种原理性的描述, 而并不是遵守严格语法的正式 C 程序. 算法开始运行时, 由于此时提取第一个图像块, 并不能计算与前一个图像块之间的相异性度量, 所以求解式(6) 计算其稀疏表示向量, 求解式(9) 计算其稀疏表示误差, 从提取第二个图像块开始便可以按照上述改进算法的流程来进行, 判断相异性度量与阈值的关系, 小于阈值, 则认为当前块和前一块近似相同, 所以两者有相同的稀疏表示向量  $\alpha_{ij}$  和表示误差  $e_{ij}^s$ , 而不必计算当前块的表示向量和误差, 缩短算法运行时间; 大于等于阈值, 则认为当前块和前一块不同, 此时将当前块的位置  $(i, j)$  记录到集合  $\delta$  中, 由于是按照连续重叠分块的方式进行提取, 所以当前块必然包含了灰度值突变的区域(边缘), 而  $\eta$  向量的非零元素位置则反映了灰度值发生改变的像素点的位置. 当稀疏编码步骤执行结束后, 集合  $\delta$  所记录位置的图像块便包含了整幅图像的边缘区域, 最后在执行“拼接”操作时, 让索引在集合  $\delta$  中的图像块跳过取平均操作而直接去“拼接”, 以保护图像边缘, 减轻去噪结果的平滑性.

### 1.3 相异性阈值 $th_d$ 的选取准则

由相异性阈值的概念并结合 1.2 小节所述改进算法的实现过程可知, 当  $th_d$  值选取过小时, 将导致有过多的图像块被认为彼此不同而执行计算, 使改进算法的运行时间接近于自适应算法, 改进效果不明显, 极端情况下取  $th_d = 0$  时, 所有的图像块都被认为是彼此不同, 都要计算稀疏表示向量和误差, 并且在“拼接”最终去噪结果时都不进行取平均操作, 此时去噪图像的边缘信息损失最小, 但改进算法的运行时间和自适应算法相比几乎没有减小, 且去噪图像会出现较为严重的人为块效应; 当  $th_d$  值选取过大时, 将导致有过多的图像块被认为是彼此相同而未执行计算, 从而使最终去噪结果的误差变大, 并且最后在拼接时过多携带边缘信息的图像块也被执行了取平均操作, 降低了对去噪图像平滑性的改善效果, 极端情况下取  $th_d = 1$  时, 所有后来提取的图像块都被认为和第一个图像块具有相同的稀疏表示向量和误差, 所以只对第一个图像块执行了计算, 此时改进算法运行时间最短, 但生成的去噪图像误差过大, 失真严重.

为了权衡运行时间和边缘信息这两者的改善效果, 要合理的选择  $th_d$  的值. 在改进算法中, 相异性阈值取为 0.6, 在 2.4 节中将说明在改进算法中为何选择  $th_d = 0.6$  以及  $th_d$  的不同取值对改进算法去噪结果的影响.

## 2 实验结果及分析

实验分为 4 组, 第 1 组实验是对不含噪声的模拟毫米波图像叠加高斯白噪声, 对比改进算法和自适应算法的去噪效果; 第 2 组实验是对获取的真实毫米波图像进行去噪来对比两者的去噪效果; 第 3 组实验是改进算法与自适应算法运行时间的对比; 第 4 组实验是研究相异性阈值的不同取值对改进算法去噪效果和运行时间的影响. 实验在 matlab R2012b 环境中进行, 实验机器内存为 8G, CPU 为 Intel I7 处理器, 图像块尺寸  $n = 8 * 8$ , 字典尺寸  $k = 256$ , 迭代次数  $J = 10$ , 相异性阈值  $th_d = 0.6$ , 实验结果如下所示.

### 2.1 第 1 组实验

#### (1) 峰值信噪比(PSNR)对比

这里按照对模拟毫米波图像舰艇(图 1), 坦克 & 汽车(图 5), 飞机(图 9) 分别叠加  $\sigma = 25$  的高斯白噪声进行去噪的方式来展示改进算法相比于自适应算法峰值信噪比的改善效果, 结果如表 1 所示.

表 1 改进算法和自适应算法得到的峰值信噪比  
Table 1 PSNR of modified algorithm and adaptive algorithm

	自适应算法	改进算法
舰艇	31.596 dB	32.306 dB
坦克 & 汽车	32.842 dB	34.083 dB
飞机	31.066 dB	32.071 dB

注:表 1 中的每个数据都是相应算法运行 5 次后,将得到的 5 个数据取平均后的结果。



图 1 无噪舰艇毫米波图像  
Fig. 1 Clean millimeter-wave image of ship

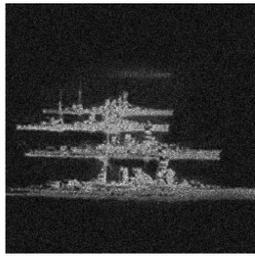


图 2 舰艇叠加  $\sigma = 25$  的高斯白噪声后的含噪图  
Fig. 2 Noisy image of ship contaminated by white Gaussian noise ( $\sigma = 25$ )



图 3 舰艇自适应算法去噪结果  
Fig. 3 De-noising image of ship by adaptive algorithm



图 4 舰艇改进算法去噪结果  
Fig. 4 De-noising image of ship by modified algorithm

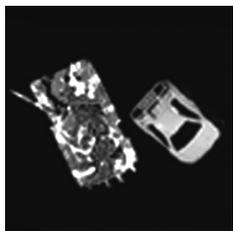


图 5 无噪坦克 & 汽车毫米波图像  
Fig. 5 Clean millimeter-wave tank & car image

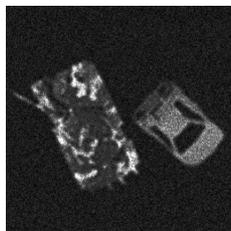


图 6 坦克 & 汽车叠加  $\sigma = 25$  的高斯白噪声后的含噪图  
Fig. 6 Noisy tank & car image contaminated by white Gaussian noise ( $\sigma = 25$ )

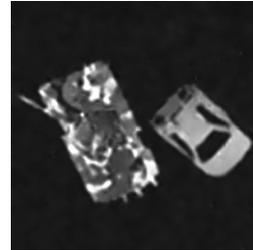


图 7 坦克 & 汽车自适应算法去噪结果  
Fig. 7 De-noising tank & car image by adaptive algorithm

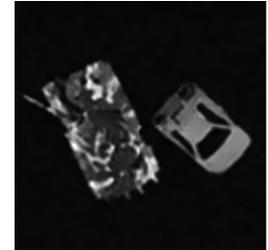


图 8 坦克 & 汽车改进算法去噪结果  
Fig. 8 De-noising tank & car image by modified algorithm



图 9 无噪飞机毫米波图像  
Fig. 9 Clean millimeter-wave airplane image

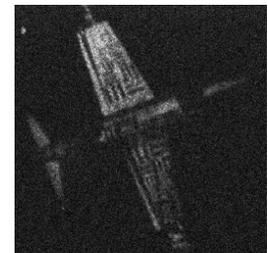


图 10 飞机叠加  $\sigma = 25$  的高斯白噪声后的含噪图  
Fig. 10 Noisy airplane image contaminated by white Gaussian noise ( $\sigma = 25$ )



图 11 飞机自适应算法去噪结果  
Fig. 11 De-noising airplane image by adaptive algorithm



图 12 飞机改进算法去噪结果  
Fig. 12 De-noising airplane image by modified algorithm

合表 1 可以看出,改进算法比自适应算法得到的 PSNR 更大,抑制噪声能力更强。

(2) 边缘保持指数<sup>[9]</sup> (Edge Preserved Index, EPI) 对比

边缘保持指数反映了去噪后图像与原始图像边缘的对比度之比,用公式表示为:

$$EPI = \frac{\sum |P_s - P_{sn}|}{\sum |P_o - P_{on}|} \quad , \quad (11)$$

其中,  $P_s$  为去噪后像素点的灰度值,  $P_{sn}$  为与  $P_s$  相邻的像素点的灰度值,  $P_o$  为与  $P_s$  对应位置处的原始图像像素点的灰度值,  $P_{on}$  为与  $P_o$  相邻的像素点的

对比图 1 至图 4, 图 5 至图 8, 图 9 至图 12, 并结

灰度值. 去噪后图像的边缘细节保持的越好,  $EPI$  的值越接近于 1.

根据式(11), 计算上述两个算法得到的去噪结果的  $EPI$ , 如表 2 所示.

表 2 改进算法和自适应算法得到的  $EPI$

Table 2  $EPI$  of modified algorithm and adaptive algorithm

	自适应算法	改进算法
舰艇	0.654	0.711
坦克 & 汽车	0.532	0.604
飞机	0.583	0.625

注: 表 2 中的每个数据都是相应算法运行 5 次后, 将得到的 5 个数据取平均后的结果.

由表 2 中的数据可以看出, 改进算法和自适应算法得到结果的  $EPI$  值都不等于 1, 这是因为改进算法和自适应算法都是从噪声图像中对原始图像进行近似估计, 而不是精确复原, 所以必然会有边缘细节信息的损失, 但与自适应算法相比, 改进算法得到的  $EPI$  更接近于 1, 说明改进算法有更好的边缘保持效果.

为了更加直观地说明改进算法相比自适应算法能保留更多的边缘信息, 下面利用边缘检测技术, 对图 1, 图 3 和图 4 进行边缘检测, 生成各自的边缘图如图 13, 图 14, 图 15 所示.

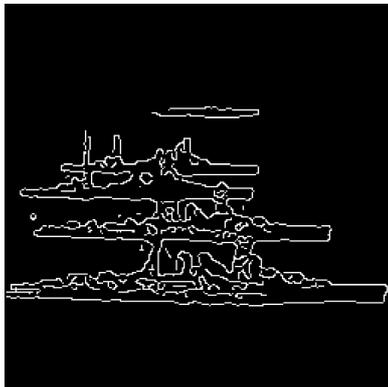


图 13 原始图像图 1 所对应的边缘图  
Fig. 13 The edge of Fig. 1

上述边缘图是由坎尼边缘检测算法(取低敏感度阈值为 0.08, 高敏感度阈值为 0.2)生成的, 对比图 14 和图 15 可以看出, 改进算法相比于自适应算法保留了更多的图像边缘(观察对比两幅图像近处的舰身区域以及远处的舰身区域).

## 2.2 第 2 组实验

这组实验采用对真实毫米波图像进行去噪的方式来对比改进算法和自适应算法的去噪效果, 图 16 为真实毫米波图像.

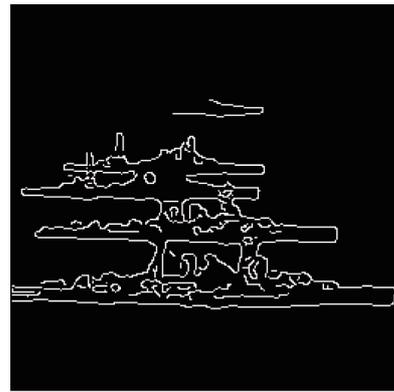


图 14 去噪结果图 3 所对应的边缘图  
Fig. 14 The edge of Fig. 3

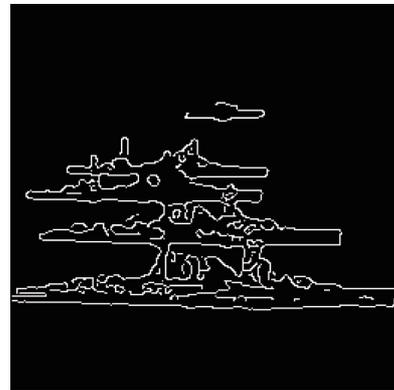


图 15 去噪结果图 4 所对应的边缘图  
Fig. 15 The edge of Fig. 4

由于真实毫米波图像所含的噪声强度是未知的, 因此首先要对噪声强度进行估计, 这是一个试探性的过程, 经过多次实验后, 当估计噪声强度  $\sigma = 30$  时, 两者所得去噪结果的相对较好, 如图 17 和图 18 所示.

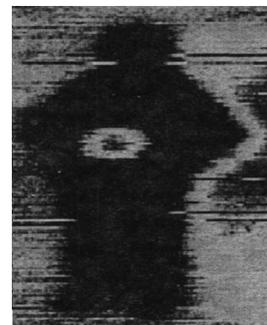


图 16 真实毫米波图像  
Fig. 16 Noisy millimeter-wave image



图 17 估计噪声标准差为 30 时自适应算法去噪结果  
Fig. 17 The result of estimating the standard deviation of the noise is 30 by adaptive algorithm  $PSNR = 25.476$  dB,  $EPI = 0.532$



图 18 估计噪声标准差为 30 时改进算法的去噪结果  
Fig. 18 The result of estimating the standard deviation of the noise is 30 by modified algorithm  $PSNR = 26.104\text{dB}$ ,  $EPI = 0.557$

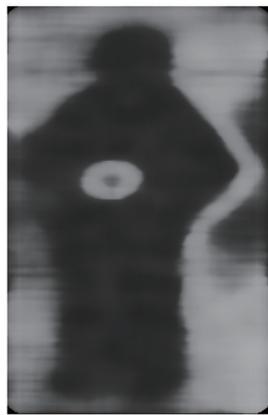


图 19 中值滤波算法的去噪结果  
Fig. 19 The result of estimating the standard deviation of the noise is 30 by median filtering  $PSNR = 24.352\text{dB}$ ,  $EPI = 0.457$

图 19 是中值滤波得到的结果,与图 17 和图 18 相比,已经出现了明显的模糊,且  $PSNR$  和  $EPI$  明显低于改进算法和自适应算法. 观察图 17 和图 18 中圆环的内边缘,图 17 相比图 18 因丢失较多边缘细节而更加模糊. 所以在对真实毫米波图像进行去噪时,改进算法相比自适应算法和传统中值滤波算法有更高的  $PSNR$  和  $EPI$ ,去噪效果更好.

### 2.3 第 3 组实验

前两组实验算法运行耗时如表 3 所示(数据由 matlab 自带的计时运行功能得到):

表 3 前两组实验中改进算法和自适应算法的运行时间对比  
Table 3 The operation time of two algorithms in experiment 1 and experiment 2

	自适应算法/s	改进算法/s
舰艇	101.264	80.762
坦克 & 汽车	83.099	70.362
飞机	66.942	50.046
真实毫米波图像	120.974	98.806

注:表 3 中的每个数据都是相应算法运行 5 次后,将得到的 5 个数据取平均后的结果.

这里主要对改进算法和自适应算法在不同噪声强度下(改变图 2 所叠加高斯白噪声的噪声强度)去噪所需的运行时间进行对比,实验结果如图 20 所示.

由表 3 以及图 20 可以看出,改进算法比自适应算法在相同噪声强度下的运行时间大约要缩短 20% 左右,这是因为一幅毫米波图像中包含边缘区域的图像块占少数,而多数都和其临近图像块近似相同,因此经过相异性阈值划分后,只需要对彼此不

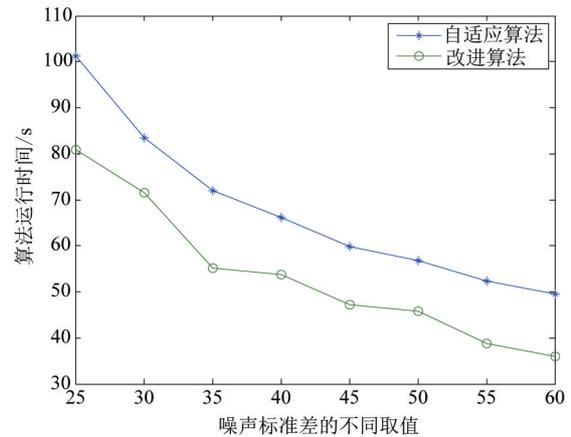


图 20 改进算法与自适应算法的运行时间随不同噪声强度  $\sigma$  的变化曲线  
Fig. 20 The relationship between operation time and  $\sigma$  for the two algorithms

同的图像块执行计算,彼此相同的图像块只对其中一个执行计算,因而压缩了算法中“同质”图像块的计算量,这就是改进算法与自适应算法相比能够减少运行时间的原因.

### 2.4 第 4 组实验

1.3 小节中,已经初步说明了相异性阈值  $th_d$  的选取准则,这里展示  $th_d$  的不同取值对改进算法的运行时间以及  $EPI$  的影响结果. 以图 2 为去噪对象来进行实验,结果如图 21 和图 22 所示.

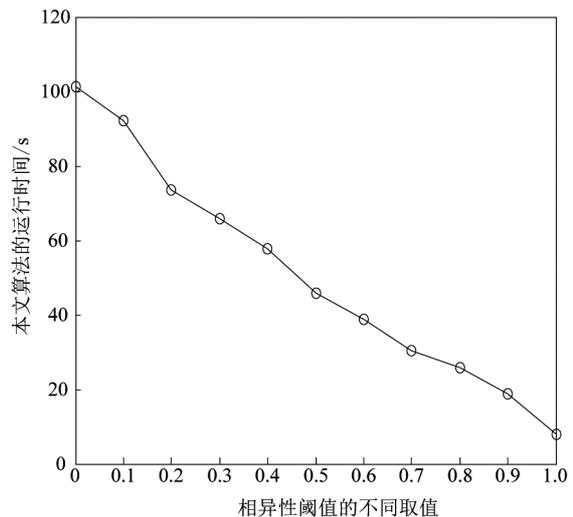


图 21 相异性阈值的不同取值对改进算法运行时间的影响曲线( $\sigma = 25$ )  
Fig. 21 The relationship between operation time of modified algorithm and  $th_d$

可以看出,两条曲线都呈波动下降的趋势,相比于自适应算法,如果为了减少运行时间,那么  $th_d$  的取值应越靠近 1 越好,但是过大的阈值导致  $EPI$  值

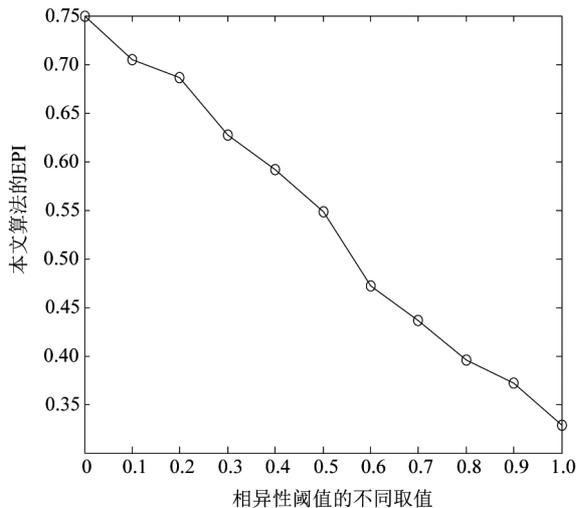


图 22 相异性阈值的不同取值对改进算法  $EPI$  的影响曲线 ( $\sigma = 25$ )

Fig. 22 The relationship between  $EPI$  of modified algorithm and  $th_d$

较小,边缘信息损失较多,并且此时参与计算的图像块数目过少,从而导致误差急剧上升,使得去噪结果产生严重的失真. 当  $th_d$  较小时,此时改进算法对  $EPI$  的改善较为理想但对运行时间的改善效果却不明显. 由此可以看出,改进算法对运行时间和  $EPI$  的改善效果是一对相互制约的因素,因此在选择  $th_d$  的取值时,不能单纯只考虑运行时间或者  $EPI$  的改善程度,要在能够向观察者展现较好的去噪图像质量的前提下权衡这两个因素. 综合考虑了不同毫米波图像分别在不同噪声强度下的多次实验结果(篇幅原因,无法逐一展示)后最终确定了相异性阈值的通用取值,取  $th_d = 0.6$ ,此时相比于自适应算法,改进算法对运行时间和  $EPI$  值的改善效果都较为理想,得到的去噪毫米波图像平滑性减轻,视觉效果良好.

### 3 结语

针对 Michael Elad 和 Michal Aharon 提出的自适应稀疏表示去噪算法的缺点,研究了基于相异性阈值的改进自适应稀疏表示去噪算法,与自适应稀疏表示去噪算法的不同之处是在算法中加入了相异性阈值判断的步骤,计算当前图像块与前一个图像块

之间的相异性度量,然后与阈值进行比较后对其进行分类处理,从而在压缩算法运行时间的同时也减少了去噪图像边缘信息的损失,减轻了去噪结果的平滑性. 最后将改进算法与自适应算法对毫米波图像的去噪效果进行了对比,验证了改进算法的有效性. 由于改进算法只进行了一次去噪,可以考虑进行多次去噪,并根据每次所产生结果的不同特点自适应地改变相异性阈值,以达到较好的去噪效果,但是这样做可能会导致算法复杂度的上升以至于仍然需要研究如何进行折中处理.

### References

- [1] Aharon M, Elad M, Bruckstein A. K-SVD: An algorithm for designing overcomplete dictionaries for sparse representation [J]. *IEEE Transactions on Signal Processing*, 2006, **54**(11):4311–4322.
- [2] Michael Elad and Michal Aharon. Image denoising via sparse and redundant representations over learned dictionaries [J]. *IEEE Transactions on Image Processing*, 2006, **15**(12):3736–45.
- [3] Wang J, Cai J F, Shi Y, et al. Incoherent dictionary learning for sparse representation based image denoising [C]. [S. l.]: 2014 IEEE International Conference on Image Processing (ICIP), 2014:4582–4586.
- [4] Li T, Wang W, Xu L, et al. Image denoising using low-rank dictionary and sparse representation [C]. [S. l.]: 2014 Tenth International Conference on Computational Intelligence and Security, 2014:228–232.
- [5] Zhang Z, Xu Y, Yang J, et al. A survey of sparse representation: algorithms and applications [J]. *Access IEEE*, 2015, **3**:1–1.
- [6] Raja H, Bajwa W U. A convergence analysis of distributed dictionary learning based on the K-SVD algorithm [C]. [S. l.]: 2015 IEEE International Symposium on Information Theory (ISIT), 2015:2186–2190.
- [7] Zhang Y Q, Xiao J S, Shuhong L I, et al. Learning block-structured incoherent dictionaries for sparse representation [J]. *Science China Information Sciences*, 2015, **58**(10):1–15.
- [8] ZHAO Jia-Jia, TANG Zheng-Yuan, YANG Jie, et al. Infrared small target detection based on image sparse representation [J]. *J. Infrared Millim. Waves* (赵佳佳, 唐峥远, 杨杰, 等. 基于图像稀疏表示的红外小目标检测算法. *红外与毫米波学报*), 2011, **02**(2):156–161.
- [9] YANG Hui-Xian, WANG Xu-Si, XIE Peng-He, et al. Infrared image denoising based on improved threshold and inter-scale correlations of wavelet transform [J]. *Acta Automatica Sinica* (杨恢先, 王绪四, 谢鹏鹤, 等. 改进阈值与尺度间相关的小波红外图像去噪. *自动化学报*), 2011, **37**(10):1167–1174.