

一种高效快速的高密度椒盐噪声消除算法

吕宗伟,唐治德,周 林,张 谦,林立森

(重庆大学输配电装备及系统安全与新技术国家重点实验室,重庆 400030)

摘 要: 本文提出了一个简单高效的消除高密度椒盐噪声的算法.在该算法中,对于图像中的每一个椒盐像素,计算以该像素为中心的窗口内非椒盐像素的平均值,并且在不同的阶段对其更新.实验结果表明,对于污染度为 10% 到 90% 的图像,本文的算法都有良好的噪声消除能力,所得到的恢复图像能够较好地保持原始图像的细节.另外,本文的算法计算时间较少,具有较大的实用价值.

关键词: 脉冲噪声; 中值滤波; 细节保持; 图像处理

中图分类号: TP391.41 **文献标识码:** A **文章编号:** 0372-2112 (2011) 08-1903-04

One Efficient and Fast Method for the Removal of High-Density Salt and Pepper Noise

LÜ Zong-wei, TANG Zhi-de, ZHOU Lin, ZHANG Qian, LIN Li-sen

(State Key Laboratory of Power Transmission Equipment & System Security and New Technology, Chongqing University, Chongqing 400030, China)

Abstract: One simple and efficient method for the removal of high-density salt and pepper noise is proposed in this paper. In the proposed method, we calculate the mean value of the non salt and pepper pixels in the window centered at the current salt and pepper pixel. Then, we update the salt and pepper pixel with the mean value at the different phases. Although the principle in the proposed method is quite simple, the experimental results show that the proposed method behaves excellently for all noise ratios, from 10% to 90%. The proposed method can preserve the details of the original image well. Furthermore, the run time for the proposed method is quite short, which makes it possible for the real applications.

Key words: impulse noise; median filter; details-preserving; image process

1 引言

图像在形成和传输的过程中往往会受到各种噪声的影响,从而使得图像的质量下降.基于此,提出了很多噪声消除的方法.其中,中值滤波(Median Filter, MF)由于算法简单和一定的细节保持能力而引起了广泛的重视^[1].但是,中值滤波会修改图像中的每一个像素,包括噪声像素和非噪声像素,从而使得恢复图像的能力有所下降^[2].为了克服这个问题,人们在中值滤波中加入噪声像素识别环节.很多方法中都大量使用各种参数(Thresholds),如:文献[3]中使用了一个参数,文献[4]中使用了4个参数,以及文献[5]中使用了很多基于模糊(Fuzzy)的规则和函数等.

文献[5]中,Chan 提出一个比较有效的方法.但是由于在该方法中,对每一个像素都是用了 39×39 的窗口,因此计算时间较长.例如:对于污染度为 70% 的 Lena 图像,该方法的计算时间大约为 1~2 个小时,而对于 90% 的污染度,计算时间更是达到了 3 个小时以上.

显然,这是不能满足实际的图像处理要求的.文献[6]中, Srinivasan 提出了一个简单的方法.它通过对 3×3 窗口的像素进行排序来得到最小值、最大值和中值,从而来判断像素是否受到污染以及计算替代像素的值.该方法计算简单,计算时间较短,是一个比较实际的方法.然而,实验发现,对于高密度污染的图像,该方法不能很好地保持原始图像的细节,因此仍然存在一定的局限性.另外,文献[7]中, Kenny 给出了一个基于自适应的模糊中值滤波算法.实验表明,对于大多数测试图像,该算法都能够得到较好的结果,尤其是当噪声密度比较高的时候.不过,该算法也存在一定的缺陷,如:仍然需要人工设置一些阈值参数以及计算时间比文献[6]中的算法多.

总体上说,目前提出的消除椒盐噪声的方法存在几个问题:一是在算法中使用过多的参数,而且这些参数的值与图像内容有关,不利于图像的自动处理;二是计算时间较长,不能满足图像处理的实时性要求;三是对高密度污染的图像,去噪效果较差,恢复图像的质量不能得到保证,容易丢失原始图像中过多的细节.

2 本文的算法

2.1 噪声检测

文献[7]中, Kenny 给出了一种利用噪声图像的直方图来识别椒盐噪声的有效方法. 通常, 对于 8 位的灰度图像, 灰度值为 0 和 255 被定义为椒盐像素.

在本文中, 对于一个图像 X , 如果某个像素 $x_{i,j}$ 的灰度值为 0 或者 255, 我们认为该像素可能是椒盐像素, 并且定义标记 F :

$$F_{ij} = \begin{cases} 1, & x_{ij} = 0 \text{ or } 255 \\ 0, & \text{otherwise} \end{cases} \quad (1)$$

即, $F_{ij} = 1$ 表示位于 (i, j) 的像素 $x_{i,j}$ 可能是椒盐像素, 否则像素 $x_{i,j}$ 是非噪声像素.

2.2 噪声滤波

首先, 对于图像 X 中的一个椒盐像素 $x_{i,j}$, 定义一个以该像素为中心的 3×3 的窗口 $S_{i,j}^3$.

$$S_{i,j}^3 = \{(k, l) : |k - i| \leq 1 \text{ and } |j - l| \leq 1, (k, l) \neq (i, j)\} \quad (2)$$

对于非中心位置的其它 8 个像素, 排除灰度值为 0 或 255 的椒盐像素, 即:

$$\Omega_{i,j}^3 = \{x_{k,l} | x_{k,l} \in S_{i,j}^3, F_{k,l} = 0\} \quad (3)$$

计算像素的均值

$$m_{i,j} = \text{mean}(x_{k,l} | x_{k,l} \in \Omega_{i,j}^3) \quad (4)$$

如果 $\Omega_{i,j}^3$ 为空, 则不计算均值.

定义一个图像 Y , 位于 (i, j) 的像素 $y_{i,j}$ 为:

$$y_{ij} = \begin{cases} m_{ij}, & F_{ij} = 1 \\ x_{ij}, & F_{ij} = 0 \end{cases} \quad (5)$$

此处, 之所以定义图像 Y , 是为了防止当前处理的椒盐像素会对其后将要处理的椒盐像素产生影响.

对于图像 Y , 如果 $F_{ij} = 1$, 则根据式(2)定义一个以像素 $y_{i,j}$ 为中心的 3×3 的窗口 $S_{i,j}^3$. 对于非中心位置的其它 8 个像素, 排除灰度值为 0 或 255 的椒盐像素, 即:

$$\Omega_{i,j}^3 = \{y_{k,l} | y_{k,l} \in S_{i,j}^3, F_{k,l} = 0\} \quad (6)$$

计算像素的均值, 并且替换像素 $y_{i,j}$

$$y_{i,j} = \text{mean}(y_{k,l} | y_{k,l} \in \Omega_{i,j}^3) \quad (7)$$

同时, 将像素 $y_{i,j}$ 标记为非噪声像素, 即: $F_{ij} = 0$.

同样, 如果 $\Omega_{i,j}^3$ 为空, 则不计算均值.

显然, 根据以上计算, 当我们对某个噪声像素进行滤波处理时, 由于窗口内的所有噪声像素都被已经处理. 因此, 此时根据窗口内的非噪声像素计算得到的均值将会更接近于原始图像的真实值. 否则, 可能会造成计算结果过大或者过小.

2.3 算法性能分析

根据式(2)~(7), 我们可以看到, 本文算法有三个显著的特点:

(1) 只对可能的椒盐像素进行计算, 而不是像文献[6]中那样, 对所有的像素进行处理, 因此待处理的像素减少很多. 同时, 这也意味, 当噪声密度增加时, 椒盐像素也会增加. 因此, 算法的计算时间也会相应增加;

(2) 对于每一个椒盐像素, 本文算法只采用均值计算, 没有涉及到文献[6]中的排序运算或者文献[7]中的模糊函数的计算. 因此, 计算量会减少很多.

(3) 本文算法不需要人工设置任何阈值参数.

综合以上, 本文算法的计算时间应该随噪声密度的增加而增加, 同时, 计算时间不会超过文献[6]和文献[7]中算法的计算时间.

2.4 算法示例分析

下面给出了一个示例图像 X .

0	114	129	0	255
255	255	115	0	0
255	255	255	0	255
255	255	0	255	157
255	0	0	255	203

对于矩阵中圆圈所标示的椒盐像素, 由于窗口内全部都是椒盐像素, 根据式(2)~(5), 将不对它进行均值计算, 图像 Y 如下.

125	114	129	133	135
114	119	115	122	112
255	115	115	136	134
114	255	0	180	157
112	193	187	183	203

根据式(6)、(7), 当我们计算圆圈所标示的椒盐像素的均值时, 由于窗口内的所有像素都标记为非噪声像素, 因此利用这 8 个像素的均值去替代该椒盐像素.

125	114	129	133	135
114	119	115	122	112
113	115	115	136	134
114	136	159	180	157
112	193	187	183	203

3 实验结果

为了说明本文算法的有效性, 我们与几个有效的算法进行了对比, 包括: 标准的中值滤波算法 SMF^[1], Sri 的算法^[6]以及 Kenny 的算法^[7].

标准测试图像 Lena, Pepper, Baboon, Jet 和 Boat 的大小为 512×512 . 窗口大小为 3×3 . 噪声为椒盐噪声, 概率为 50%, 即灰度值最大和最小的椒盐噪声出现的概率是一样的. 椒盐噪声密度从 10% 变化到 90%, 每隔 10% 测试一次. PSNR (Peak Signal-to-Noise Ratio) 值用来客观估计恢复图像的质量, PSNR 值越大, 表示恢复图像与原始图像接近.

表 1~5 给出了 5 个图像的测试结果. 从中我们可以看到, 对于各种密度的噪声, 本文的算法要明显好于 Sri 的算法. 而对于 Kenny 算法, 当噪声密度比较低的时候

候,例如:噪声密度小于 50%,本文算法的平均计算结果要优于 Kenny 算法.而当噪声密度比较高的时候,本文算法的平均计算结果基本上接近于 Kenny 算法.

表 1 图像 Lena 的测试结果

Noise	PSNR for image ‘Lena’ (dB)			
	SMF	Sri	Kenny	Ours
10%	33.08	41.53	42.15	42.29
20%	28.58	37.48	38.45	39.25
30%	23.43	34.73	35.87	36.91
40%	19.00	32.18	33.93	34.99
50%	15.39	30.21	32.03	33.35
60%	12.28	28.15	30.57	31.71
70%	10.03	25.68	28.97	29.80
80%	8.12	23.16	27.15	27.38
90%	6.63	19.62	24.54	24.85

表 2 图像 Pepper 的测试结果

Noise	PSNR for image ‘Pepper’ (dB)			
	SMF	Sri	Kenny	Ours
10%	33.12	31.46	33.56	40.27
20%	28.59	30.75	31.84	36.99
30%	23.55	30.07	30.38	34.75
40%	18.77	28.95	29.04	33.32
50%	15.08	27.87	27.77	31.67
60%	12.24	26.28	26.62	30.09
70%	9.83	24.39	25.43	28.20
80%	8.01	22.30	24.12	25.66
90%	6.50	18.69	22.37	22.39

表 3 图像 Baboon 的测试结果

Noise	PSNR for image ‘Baboon’ (dB)			
	SMF	Sri	Kenny	Ours
10%	23.11	28.93	33.12	31.88
20%	22.01	26.82	29.58	28.89
30%	19.97	25.16	27.61	27.01
40%	17.25	23.75	26.04	25.56
50%	14.42	22.37	24.69	24.44
60%	12.00	21.14	23.47	23.31
70%	9.90	19.89	22.26	22.19
80%	8.11	18.53	21.04	21.07
90%	6.69	16.89	19.53	19.76

表 4 图像 Jet 的测试结果

Noise	PSNR for image ‘Jet’ (dB)			
	SMF	Sri	Kenny	Ours
10%	32.20	31.15	33.14	39.99
20%	27.75	30.47	32.43	36.76
30%	23.04	29.46	31.72	34.37
40%	18.20	28.27	31.02	32.76
50%	14.62	27.01	30.19	31.03
60%	11.80	25.26	29.34	29.17
70%	9.51	23.48	28.27	27.24
80%	7.59	21.25	27.25	24.85
90%	6.13	18.45	25.21	21.70

表 5 图像 Boat 的测试结果

Noise	PSNR for image ‘Boat’ (dB)			
	SMF	Sri	Kenny	Ours
10%	30.41	38.50	40.07	39.57
20%	27.11	34.71	36.31	36.30
30%	22.79	32.06	33.88	34.10
40%	18.44	29.84	32.06	32.45
50%	14.92	27.88	30.50	30.74
60%	12.15	25.98	29.20	29.05
70%	9.94	23.77	27.71	27.23
80%	8.03	21.44	26.12	25.02
90%	6.54	18.54	23.82	21.82

图 1 给出了当噪声密度为 80% 时,各种算法对 5 个测试图像的恢复结果.其中,在每个图例中,从左到右依次为:原始图像,SMF 算法的恢复图像,Sri 算法的恢复图像,Kenny 算法的恢复图像和本文算法的恢复图像.

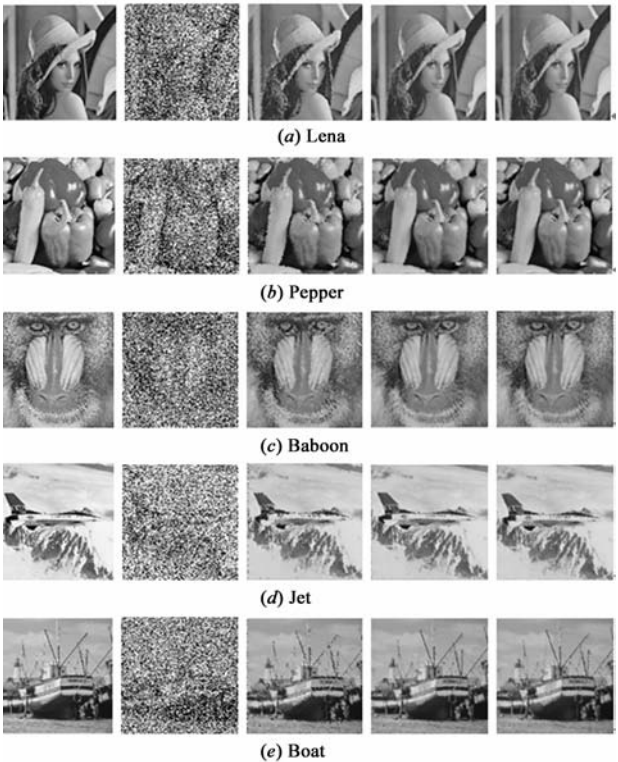


图 1 噪声密度为 80% 时 SMF 算法、Sri 算法、Kenny 算法和本文算法的恢复图像

从图 1 中可以看到,Sri 算法、Kenny 算法和本文算法都能够较好地恢复原始图像.但是在图像的细节保持能力上,Kenny 算法和本文算法明显要好于 Sri 算法.另外,我们也可以观察到,对于这 5 个测试图像,Kenny 算法和本文算法在图像的恢复以及细节保持能力上几乎没有非常明显的差异.同时,我们也对其他一些标准图像进行了测试.实验结果也表明,对于高密度噪声,本文算法能够与 Kenny 算法得到类似的计算结果.

从计算时间上来看,如前所述,在 Sri 算法中,由于要对所有的像素采用相同的运算,包括:排序和中值计算,所以其运行时间基本上与噪声密度无关.在 Kenny 算法中,只对噪声像素进行运算,所以运行时间与噪声密度有关.也就是说,噪声密度越高,计算时间越长.同时,Kenny 算法的运算包括:中值计算和模糊函数计算等,其计算时间要多于 Sri 算法.而在本文的算法中,由于只对噪声像素进行运算,因此其运算时间随噪声密度的增加而增加.另外,由于本文算法的运算只包括简单的均值计算,从而使得算法的计算时间应该小于 Sri 算法和 Kenny 算法.对于 5 个标准测试图像的平均计算时间如图 2 所示.

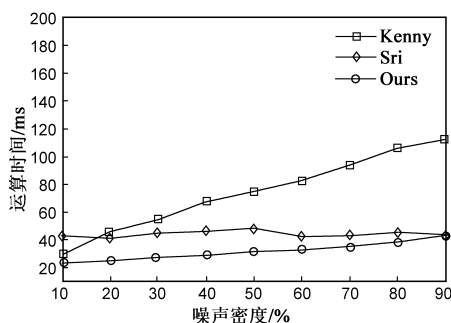


图2 测试图像的平均计算时间

4 结语

针对目前已经提出的消除椒盐噪声的算法的缺陷,如:需要设置参数,较长的计算时间以及对高密度噪声的图像恢复效果较差等特点,我们在本文给出了一个简单高效的算法.该算法不需要人工设置任何参数,便于实现图像的自动处理.计算时间比大多数算法都少,具有较强的实时性.另外,最为重要的是,对于高密度的椒盐噪声,本文的算法能够很好地恢复图像,原始图像的大部分细节都能够得到有效的保持.实验结果也表明了该算法的计算结果比 Sri 算法要好.而对于高密度噪声,其计算结果几乎与 Kenny 算法没有明显的差异.

致谢 衷心感谢 Kenny Kal Vin Toh 提供论文所需的实验数据.

参考文献

- [1] DRK Brownrigg. The weighted median filter[J]. Commun Ass Comput Mach, 1984, 27(8): 807 - 818.
- [2] Z Wang, D Zhang. Progressive switching median filter for the removal of impulse noise from highly corrupted images[J]. IEEE Trans Circuits Sys, 1999, 46(1): 78 - 80.
- [3] T Sun, Y Neuvo. Detail-preserving median based filters in image processing[J]. Pattern Recognit Lett, 1994, 15(4): 341 - 347.
- [4] E Abreu, M Lightstone, S KMitra, K Arakawa. A new efficient approach for the removal of impulse noise from highly corrupted images[J]. IEEE Trans Image Processing, 1996, 15(6): 1012 - 1025.
- [5] R H Chan, C-W Ho, M Nikolova. Salt and pepper noise removal by median type noise detectors and detail preserving regularization[J]. IEEE Trans Image Processing, 2005, 14(10): 1479 - 1485.
- [6] K S Srinivasan, D Ebenezer. A new fast and efficient decision-based algorithm for removal of high-density impulse noise[J]. IEEE Signal Processing Letters, 2007, 14(3): 189 - 192.
- [7] Kenny KVT, Nor A M Isa. Noise adaptive fuzzy switching median filter for salt and pepper noise reduction[J]. IEEE Signal Processing Letters, 2010, 17(3): 281 - 284.

作者简介



吕宗伟 男,副教授,1973 年生于重庆.1995 年毕业于天津大学自动化系,获学士学位,1999 年毕业于天津大学自动化系,获硕士学位,2002 年毕业于上海交通大学电子系,获博士学位.主要研究方向为图像处理、集成电路设计等.

E-mail: luzongwei@hotmail.com



唐治德 男,1958 年生于重庆市,1982 年毕业于哈尔滨工业大学,获学士学位;之后在重庆大学获硕士和博士学位.现为重庆大学教授,从事电力电子技术和图像处理方面的研究和教学工作.