

# 拉普拉斯差分算子同步降噪空间滤波器设计

王众 叶华 杨飞 齐正霞

(哈尔滨工程大学 自动化学院, 哈尔滨 150001)

**摘要** 为了在继承拉普拉斯算子高实时性的前提下提高图像分割质量,设计了一个可与拉普拉斯算子差分计算同步进行滤波工作的空间滤波器。以被检测边缘的连通域为单位,建立连通域点集及收录二值化数据的树集,引入面积阈值进行滤波。运用游程编码的思想作为树集设计逐行扫描的数据录入方法,使数据录入与拉普拉斯差分运算同步进行,以保证高实时性。边缘检测时通过降低差分计算的颜色阈值获取更为完整的检测结果,同时利用空间滤波器去除由颜色阈值的降低所引发的大量小面积噪声。试验结果表明:引入空间滤波器的拉普拉斯算子可在保证高实时性的前提下获取低噪声的且更加完整的边缘检测结果。同步空间滤波器的引入,可使原算法在低颜色阈值条件下获得高质量的图像分割结果,且该滤波器的行扫描数据录入方式可以保证高实时性。

**关键词** 拉普拉斯算子; 空间滤波器; 边缘提取

**中图分类号** TP 391.41

**文章编号** 1007-4333(2006)05-0107-06

**文献标识码** A

## Design of synchronization dimensional structure filter for Laplace operator

Wang Zhong, Ye Hua, Yang Fei, Qi Zhengxia

(Automation College, Harbin Engineering University, Harbin 150001, China)

**Abstract** In order to improve the quality of Laplace operator image segmentation at a high real time capability this paper presented a synchronization dimensional structure filter for Laplace operator. After constituting interconnected domain volume, a square threshold is introduced. Using the Run-Length coding method, implementing the synchronization of volume data-recording and Laplace difference, the filter keeps a high real time capability. The synchronization filter filtrates the thin square noise when the color threshold is depressed to obtain a more intact detection. The result indicates that the dimensional structure filter edge method can obtain a more intact and less noise image segmentation. The introduction of the synchronization dimensional structure filter makes the Laplace operator improve the segmentation quality with a high real time capability by the original arithmetic.

**Key words** Laplace operator; dimensional structure filter edge; extraction

图像分割是图像处理和计算机视觉领域低层次视觉中的主要问题,同时也是一个经典难题,特征提取和模式识别等都依赖于图像分割的质量。计算快捷、精度高、适应性强的降噪分割算法一直是图像降噪分割问题研究的核心。传统的方法有领域平均法<sup>[1]</sup>、倒数梯度加权法<sup>[1]</sup>、线性插值法<sup>[2]</sup>和中值滤波法<sup>[3]</sup>,近年来出现了自适应彩色图像滤波方法<sup>[4]</sup>和基于总体变差模型的滤波方法<sup>[5]</sup>等。小波域内的图像去噪是另一大热点,包括小波域内的软门限

去噪<sup>[6]</sup>、基于双 Haar 小波的软门限去噪<sup>[7]</sup>,以及基于随机几何形态学的小波图像去噪<sup>[8]</sup>方法。近期的研究中,文献[9]采用双阈值与空间相位编码相结合的方式补救颜色梯度变化不大的边缘;文献[10]以点对不同阈值集的从属关系判断不同阈值集之间的空间连通关系,从而根据不同的空间特征设置动态阈值;文献[11]以颜色相似度为依据建立最优路径,在最优路径的基础上加入点与种子相似度的判断用以去除噪声。从目前的研究进展来看,在颜色

收稿日期: 2006-03-22

作者简介: 王众,博士研究生,主要从事导航自动化研究, E-mail: wang\_ren3y@yahoo.com.cn

阈值的基础上引入空间特征判断已经成为首选的降噪手段,此外多个算法的融合与集成可以克服单个算法的局限性。图像分割领域中的分割质量与高实时性一直是一对矛盾。高质量算法往往复杂度也很高,计算量的猛增导致其实时性降低;而低复杂度高实时性的算法产生的噪声很大,导致计算结果无法满足实际需要。如果能在一个高实时性算法上加入一个高实时性的空间滤波器,就能在不大幅降低实时性的前提下提高计算结果的质量。拉普拉斯算子是边缘提取中最简单的差分算子之一,但其噪声也很大。本研究拟在拉普拉斯算子算法中嵌入一个空间滤波器,希望能在继承原算法高实时性的前提下,提高算法分割结果的质量。

## 1 算法

拉普拉斯算子是以逐行扫描的方式进行差分计算的,如果能在进行差分的同时将数据收录到滤波器中就可以避免2次扫描而节省大量时间。Rosenfeld曾经指出,对目标像素的行程进行标号要快于对每个像素本身进行标号<sup>[12]</sup>,而目前也有将这种独特的处理方式运用在并行计算机上的研究<sup>[13]</sup>。

本文算法首先建立一个数据结点地址扫描行,利用这个扫描行对由拉普拉斯边缘算子差分出的当前行像素的数据空间分布进行记录,在算法对下一行像素进行计算的时候,根据数据结点地址扫描行中记录的上一行的信息对当前处理的结点做出空间分布的判断,并进行相应的树分支链接与合并;然后将得到的链表结构数据存储到数据树上。通过对数据树的遍历访问得知每个分支的大小及有无再增长可能性;通过分枝的参数与阈值的比较确定这个分支是否属于噪声,从而达到边差分边滤波的目的。

### 1.1 基本数据结构

```
typedef struct node{
    int x, y; 数据点坐标值;
    int num; 数据点个数;
    int date, changed, line; 分枝增长可能性;
    node * up, down, left, right; 数据链接指针;
    node * tail; 尾部数据;
}node
```

### 1.2 差分滤波点集定义域

设图1为要处理的数据图像,U型线条是有待提取的数据线条,右侧的单个黑点是要去除的噪声,A至I为每个黑点的标号。

为了实现差分与滤波同步,必须制定合适的定义域。为了实现单点步进、逐行扫描,按照差分—滤波—差分的方式进行计算,取滤波定义域为差分定义域中已完成差分计算的最大子集。由此给出滤波定义域:

设被处理图片由  $P \times Q$  个像素点阵组成。

若当前差分点  $E \{(U, Q) | 0 \leq U < P\}$ , 则定义域点集  $M$  为

$$\{(X, Y) | X = U - N_1, Y = Q - N_2, N_1 = 0, 1, N_2 = 0, 1\};$$

若当前差分点  $E \{(0, V) | 0 \leq V < Q\}$ , 则定义域点集  $M$  为

$$\{(X, Y) | X = 0 + N_1, Y = V - N_2, N_1 = 0, 1, N_2 = 0, 1\};$$

若当前差分点  $E \{(U, 0) | 1 \leq U < P - 1\}$ , 则定义域点集  $M$  为

$$\{(X, Y) | X = U + N, Y = 0, N = -1, 0, 1\};$$

若当前差分点  $E \{(U, V) | 1 \leq U < P - 1, Q > 0\}$ , 则定义域点集为

$$\{(X, Y) | X = U - N_1, Y = V - N_2, N_1 = 0, 1, N_2 = 0, 1\} \cup \{(X, Y) | X = U + 1, Y = V - 1\}。$$

### 1.3 点对点的链接规则

根据拉普拉斯算子的特性设置好合适的颜色梯度阈值,就可以把黑点提取出来。为了将检测出的数据点按照连通域点并入同一分支的原则链入树型结构,必须对差分出的数据点进行链接对象判断。为此制定链接规则,即捕捉到的数据点链入其定义域中按顺时针顺序第1个被检测到的数据点尾部。如果其定义域中没有数据结点,这个点将作为分支的首结点存入树中。在图1中由于A、B、C的定义域中不含数据点,所以它们将作为首结点链入树中,而D点将链接到A的尾部。I点的定义域中有H、G 2个数据点,由于按顺时针H先于G被检测到,所以I链接到H的尾部。链接规则定义如下:

设  $\forall$  差分点  $E$ , 定义域点集  $M, M \in E$ 。取颜色阈值  $N$ , 如果  $f(E) \geq N$ , 则  $E_r = 255, E_b = 255, E_g = 255$ ; 如果  $f(E) < N$ , 则  $E_r = 0, E_b = 0, E_g = 0$ ; 如果  $f(E_1) \geq N$  且  $f(E_2) < N$ , 则做关系式  $nLink(E_1, E_2) : E_1 \text{ down} = E_2, E_2 \text{ up} = E_1$ 。  $f(E)$  为点  $E$  处的8方向拉普拉斯差分运算。

### 1.4 分支与分支的链接规则

定义存放在树型结构上的连通域点集为分支。若在扫描第  $L$  行时  $f(E_L) \geq N, E_L$  与当前定义域内

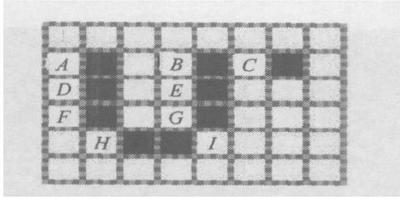


图 1 算例

Fig. 1 Example of calculation

的点均不符合点对点的链接规则,且在扫描第  $L+1$  行时  $\forall E_2$  与  $E_1$  符合点对点的链接规则,则称  $E_1$  为分支  $T = \{E_1, E_2, \dots\}$  的首结点。

在按照点对点的链接规则计算完  $F$ 、 $G$  行后,树型结构中出现了 3 个分支,它们是  $A-D-F$ 、 $B-E-G$  与  $C$  分支。从图 1 可见,  $A-D-F$  与  $B-E-G$  本应属于同一分支,但在没有扫描到  $H$ 、 $I$  行时,无法判断它们是否属于同一连通域;所以在算法规则中必须加上分支与分支合并的判断条件来合并属于同一连通域的分支。在进行  $H$ 、 $I$  行扫描时,按照点对点链接规则,  $H$  链接到  $F$ ,  $I$  链接到  $H$ ;但对  $I$  的定义域检测不能到此结束,要继续判断  $I$  是否还属于其他分支,如果还属于其他分支,则说明 2 个  $I$  之前不连通的分支由于  $I$  的出现而联通,所以要把这 2 个分支合并。具体操作为,将后检测到的分支的首结点从树中剪下,再链接到最先检测到的分支的尾部。于是在扫描完  $I$  点时,树中的分支为  $A-D-F-B-E-G$  与  $C$ 。给出的分支对分支的链接规则如下:

设  $\forall$  差分点  $E$ ,  $M$  为定义域点集,若  $\exists$  分支  $T_1$   $M \neq \emptyset$ ,使  $T_1 \cap E \neq \emptyset$ ,且  $\exists$  分支  $T_2$   $M \neq \emptyset$ ,使  $T_2 \cap E \neq \emptyset$ ,则作  $nlink(E, head(T_2))$ 。  $head(T_2)$  为  $T_2$  所在分支的首结点。

### 1.5 分支有无再增长可能性定义

根据以上的规则和定义,算法的基本功能已经完成;但是为了减少对硬件资源的消耗,加快计算机的运行速度,必须将噪声点及时清理掉。在检测到  $D$ 、 $E$  行时,树中  $C$  分支就已经停止了生长。如果  $C$  分支小于面积阈值,就可以把它当成噪声分支进行清除。为了直观地反应出滤波效果,在将噪声分支从树中删除的同时将噪声分支中记录的图片上的位置点涂黑,这个过程也就是滤波的过程,所以判断有无增长可能性是用来区分噪声分支的前提。为此给出分支有无增长可能性的定义:

设当差分前处理行为第  $L_1$  行,且分支  $T$  中的最近数据更新行为第  $L_2$  行,若  $L_1 = L_2$ ,则  $T$  无增

长可能性,记:  $P(T) = 0$ ;若  $L_1 < L_2$ ,则  $T$  有增长可能性,记:  $P(T) = 1$ 。

加入此判断后,在  $D$ 、 $E$  行扫描结束时,树中分支为  $A-D$ 、 $B-E$ 。以后的计算中减少了  $C$  分支对内存的消耗。

### 1.6 双树结构

为了进一步提高运算速度及节省计算机资源,算法采用 2 个树集来存贮数据。

缓冲树集的定义:  $\{T | P(T) = 1\}$ ;

寄存树集的定义:  $\{T | P(T) = 0, m_T \leq N\}$ 。

$N$  为设定的阈值,  $m_T$  表示分支  $T$  中结点数。连通域先进入缓冲树,只有符合要求才能进入寄存树;而每次滤波扫描都是在缓冲树中进行的。缓冲树中连通域的大小受到严格限制,小连通域在几次扫描之后就会被剔除,而大于阈值的连通域一旦失去了增长的可能性又会被剪切到寄存树中。

### 1.7 数据提取流程

流程如下:

Step1 遍历当前行结点;

Step2 读入当前结点的数据;

Step3 拉普拉斯差分运算;

Step4 单点数据点处理;

Step5 滤波;

Step6 行扫描是否结束,如果是,执行 Step7,如果不是,跳至 Step1;

Step7 地址数组更新;

Step8 是否还有下一行,如果是,读入下一行数据并跳至 Step1,如果不是,执行 Step9;

Step9 结束。

## 2 试验

试验环境为 Windows XP 操作系统, 2.0 GHz 奔腾处理器,内存为 256 MB。使用 VC6.0 进行算法开发。使用 Matlab 的 COM 组件进行数据统计,以计量不同时间算法对内存的消耗情况。在图 2 中,(a)为 Lena 原图,图片像素数 65 536 个。(b)为二值化后的 8 方向拉普拉斯算子检测结果,选取的颜色梯度阈值为 70,耗时 422 ms。由于阈值较高,它的主体边缘受到严重损伤。(c)为降颜色梯度阈值为 20 的 8 方向拉普拉斯算子边缘提取结果,耗时 468 ms。由于降低了颜色阈值,主体边缘大部份被保留下来;但低阈值导致噪声激增。(d)为颜色梯度阈值 20,加入空间滤波器,取面积阈值为 6 的边缘

提取结果,耗时 500 ms。加入面积限制条件后,绝大部分噪声被剔除。由图 3(b)可以看出,无滤波处理算法得到的结果中存在大量小面积检测边缘,其中很大一部分是由汇聚在广场前的车流及广场上的人群造成的,而人流与车流对于景象匹配运算将会造成很大干扰。加入空间滤波器后,显著降低了这种噪声(图 3(c))。

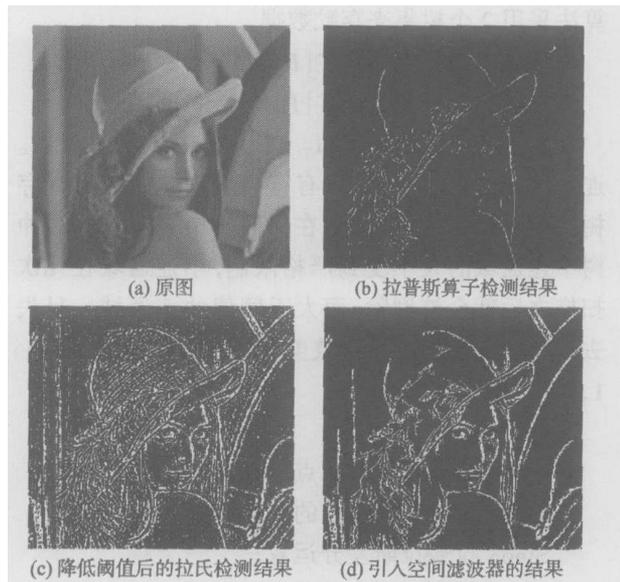


图 2 Lena 图像处理结果

Fig. 2 Comparison among Lena and segmentation: (a) original picture; (b) Laplace segmentation; (c) Laplace segmentation with low depressed threshold; (d) Laplace segmentation with dimensional filter

图 4 示出颜色阈值取 40 时,让面积阈值成倍增长的破坏性试验结果(原图见图 2(a))。可见要想取得比较理想的滤波效果,需设定适当的面积阈值。图 4(a)为面积阈值取 0 时的情况,(b)的面积阈值为原图像素面积的 0.001 5%,(c)为 0.000 3%,(d)为 0.000 6%;但是不能就此判定 0.001 5%就是所有图像的最佳面积阈值。引入面积阈值的目的在于划分噪声与主体数据在面积上的界限。由于不同图像的噪声与主体边缘的面积界限并不一样,所以面积阈值同颜色梯度阈值一样需要针对不同图像而定。面积阈值过小,一些噪声去不掉;面积阈值过大又会伤害主体边缘。

### 3 结果分析

组成图片的像素数与待提取的线条数量及噪声的多少将决定算法需消耗的时间。寄存树中结点的数量反映了待提取的线条数量。缓冲树中的结点是扫描过程中数据与噪声的混合体,然而,一旦缓冲树

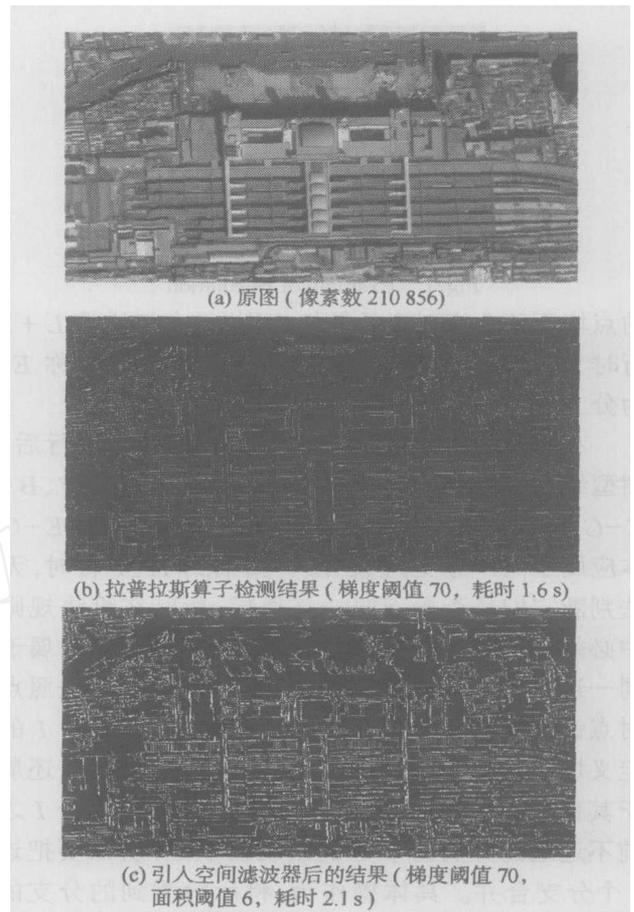


图 3 北京火车站航拍图像处理结果

Fig. 3 Processed airscape images Beijing railway station: (a) Original picture with 210 856 pixels; (b) Laplace segmentation (color durance: 70, time spend: 1.6 s); (c) Laplace segmentation with dimensional filter (color durance: 70, square durance: 6, time spend: 2.10 s)

中的分支被确认是噪声或是主体边缘数据,就会被剔除出缓冲树。这也是缓冲树的体积得到控制的根本原因。比较同一幅图像的缓冲树与寄存树的大小,可以得出一个对算法效率的参照,即为了提取寄存树中的数据,在提取过程中到底要被寄存树中的不确定数据消耗掉多少内存。本文中每个结点是 8 个字节,256 MB 内存中共可容纳  $256 \times 1024 \times (1024/8) = 33\,554\,432$  个结点。图 2(a)内容比较单一,对应的缓冲树中的连通域峰值并未超过 63 个(图 5(b)),而对应的寄存树中的连通域数最终达到 43 个(图 5(a));即为了提取 43 个有用的数据,对内存消耗的峰值为 62。图 3(a)内容比较丰富,虽然其缓冲树中的连通域数较图 5(b)激增,但峰值也没有超过 130 个(图 6(b)),远远小于寄存树中的最终峰值(380 个,见图 6(a))。对内存消耗的峰值没有需要提取的数据多,说明图像中面积大的连通域

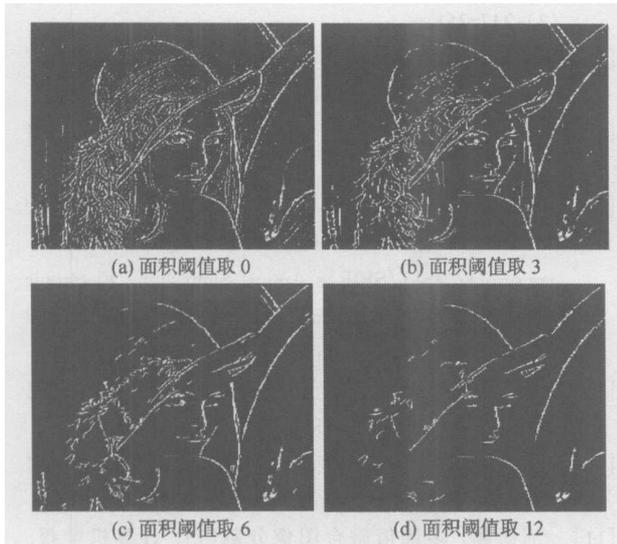


图 4 面积阈值不同时 Lena 图像的空间滤波结果 (颜色阈值 40)

Fig. 4 Change square threshold under a constant color threshold

较少;同时也证明对于需要频繁遍历访问的缓冲树来说,本文中提出的双树结构可以有效降低缓冲树的体积,从而提高运算效率。另外,2 幅图像的滤波器算法耗时分别较无滤波器时增加 6.84% 和 23.8%,说明算法的效率会随着图像复杂度的增加而下降。

由于本滤波器以拉氏算子差分结果为前提,所

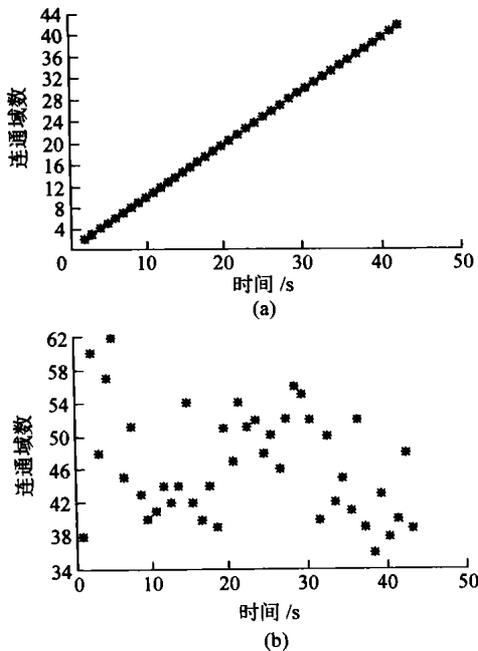


图 5 Lena 图像寄存树 (a) 和缓冲树 (b) 中连通域数量随时间的变化

Fig. 5 Variety of date in consignment tree (a) and cushion tree (b) for Lena

以不可避免地继承了拉氏算子差分的缺点:1)对于颜色梯度变化不明显的边界依然检测不出来;2)如果阈值设定得过低,提取出来的边界毛刺比较多。在对数据提取质量要求较高的场合,对于缺陷 1),可用高阶样条曲线来拟合缺失的边缘;对于缺陷 2),可用曲线的平滑算法去除毛刺。从整体效能来看,本算法比较适合于对实时性要求很高且对细节的要求并不高的图像处理,如导航制导中的图像匹配。

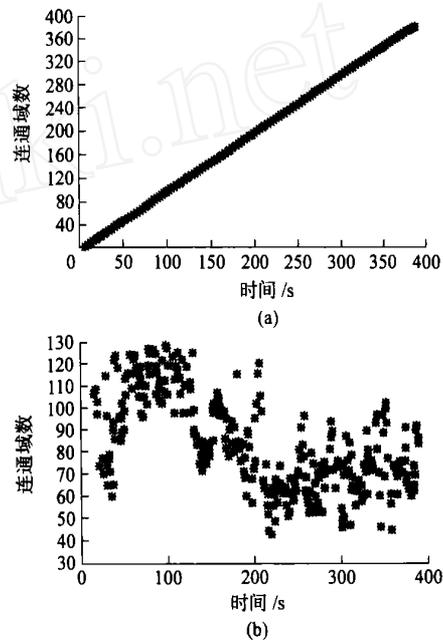


图 6 北京火车站航拍图像寄存树 (a) 和缓冲树 (b) 中连通域数量随时间的变化

Fig. 6 Variety of date in consignment tree (a) and cushion tree (b) for Beijing railway station

#### 4 结束语

图像分割质量的好坏通常由人类的视觉习惯来衡量。多数图像分割算法本质上都是在试图建立更加接近于人类视觉的数学模型。人眼不仅靠颜色的梯度变化来识别物体的边缘,空间特征也是其识别边缘的主要依据。虽然面积是很基本的空间特征,但从本文的试验结果来看,面积阈值的引入能够显著提高图像分割质量,所以对人类视觉习惯的研究将有助于推动图像分割算法的进步。另外,设计由计算机运行的算法时,要兼顾计算机的运行效率,才能提高算法的实时性。由于图像分割运算具有数据处理量大的特点,算法实时性的高低主要取决于数据录入内存的效率以及对内存的消耗量。综合考虑上述问题,本文中提出的同步空间滤波器既保持

了高实时性,又降低了噪声。

同步空间滤波器采用行扫描数据录入方式,从像素的连接关系入手,把面积小的连通域归为噪声。对于一幅没有受到强噪声干扰的图像,噪声普遍具有面积小的特征,所以根据这一特征可以筛选出绝大多数噪声;但是面积小的连通域不一定是噪声,采集到的边缘数据有时会因颜色阈值设置过高受到损伤而发生断裂,这种情况下显然不应把断裂下来的连通域当做噪声来处理。对于这种破损数据的鉴别甚至修复尚有待于进一步的研究。

### 参 考 文 献

- [1] 傅德胜,寿益禾. 图形图像处理学[M]. 南京:东南大学出版社,2002:188-190,192-204
- [2] Gonzalez R C, Woods R E. 数字图像处理[M]. 阮秋琦,译. 北京:电子工业出版社,2003:93-97
- [3] Castleman K R. 数字图像处理[M]. 北京:电子工业出版社,2002:204-206
- [4] 蒋刚毅,郁梅,郁伯康. 自适应彩色图像滤波新方法[J]. 科学通报,1999,44(18):1937-1942
- [5] 肖亮,吴慧巾,韦志辉,等. 基于总体变差模型的数字滤波器设计及其性能研究[J]. 信号处理,2003,19(3):247-251
- [6] Donaho D L. De-noising by soft thresholding[J]. IEEE Transactions on Information Theory, 1995,41:613-627
- [7] 王欣. 基于双 Haar 小波的辑门限去噪[J]. 山东大学学报,2003,19(3):35-37
- [8] Voloshynovskiy S, Koval O, Pun T. Wavelet-base image denoising using non-stationary stochastic geometrical image priors. ISIT/SPIE's Annual Symposium, Electronic Imaging 2003: Image and Video Communications and Processing, San Clara: ISIT/SPIE, 2003:20-24
- [9] 郝罡. 基于边缘提取的无人机定位方法研究[J]. 测控技术,2004,23(7):77-78
- [10] 潘建江. 基于模糊连接度的图像分割及算法[J]. 软件学报,2005,16(1):68-74
- [11] 刑研超. 多阈值融合图像分割[J]. 计算机学报,2004,27(2):253-256
- [12] Rosenfeld A, Pfaltz P. Sequential operations in digital picture processing[J]. Journal of the Association for Computing Machinery, 1966, 12: 474-493
- [13] Choudhary A, Thakur R. Connected component labeling on coarse grain parallel computers:an experimental study[J]. Journal of Parallel and Distributed Computing, 1994, 20(1):78-83