

文章编号: 1005-0523(2009)01-0052-06

# 分水岭算法在胃上皮内肿瘤图像分割中的应用

张红斌<sup>1</sup>, 李广丽<sup>1</sup>, 刘遵雄<sup>1</sup>, 黄一联<sup>2</sup>

(1. 华东交通大学 信息工程学院, 江西 南昌 330013; 2. 北京大学 软件学院, 北京 100871)

**摘要:** 图像分割是图像识别的关键, 笔者曾经分别采用松弛迭代<sup>[1]</sup>和 K 均值聚类方法对胃上皮内肿瘤图像进行分割, 实验表明这些算法对粘连严重的图像分割效果很差, 故本文应用分水岭分割算法 Vincent 和 Inver 对粘连情况不同的多类胃上皮内肿瘤图像进行了图像分割实验. 实验结果表明: 对于粘连较少的细胞图像, 这两种算法都能较好地分离出目标细胞, 但对于粘连严重的细胞图像, Inver 算法的分割效果比 Vincent 要好, 但 Inver 算法容易出现过分割现象.

**关键词:** 胃上皮内肿瘤; 图像分割; 分水岭算法; Vincent; Inver

**中图分类号:** TP 391.4

**文献标识码:** A

随着计算机、模式识别、数字图像处理等技术的发展, 基于医学数字图像的医疗诊断已经成为病理诊断中的一个非常重要的环节. 因为, 医学图像中不同组织、病灶体在图像上表现为不同的边界、纹理及光密度特征. 病理医生可凭借经验进行肉眼识别, 或者借助医学图像软件进行自动识别. 后者可以获取更高的工作效率和较高的识别正确率, 极大地方便了病理医生的病理诊断工作. 医学图像软件由图像采集、预处理、图像分割和分类识别四个部分组成, 图像分割是系统提高识别率的关键, 笔者正参与一个“医学图像处理软件”项目, 并在文献 1 中尝试使用松弛迭代分割算法<sup>[1]</sup>对胃上皮内肿瘤图像进行分割, 实验结果表明: 对于部分无粘连的细胞图像, 该算法可以获得较好的分割效果, 而粘连严重的细胞图像该算法则无能为力, 故本文重点研究分水岭算法在胃上皮内肿瘤图像分割中的应用.

## 1 分水岭分割算法的设计

算法描述如下: 令  $M_1, M_2, \dots, M_r$  表示图像  $g(x, y)$  中局部最小点的坐标集合, 即低洼. 令  $C(M_i)$  表示与局部最小值  $M_i$  相关联的集水盆地<sup>[2]</sup>.  $\min$  和  $\max$  代表  $g(x, y)$  中灰度最小值和最大值. 令  $T[n]$  表示坐标  $(s, t)$  的集合, 其中  $g(s, t) \leq n$  即灰度小于等于阈值的所有像素

$$T[n] = \{(s, t) \mid g(s, t) \leq n\} \quad (1)$$

几何上,  $T[n]$  集合中的点均位于平面  $g(x, y) = n$  的下方. 随着水位从  $n = \min + 1$  到  $n = \max + 1$  不断升高, 图像中的地形会被水逐渐漫过. 在水漫过地形的每一阶段, 算法都需要知道处在水位之下的点的数目. 假设  $T[n]$  中的坐标处在  $g(x, y) = n$  平面之下, 并被标记为黑色, 而其它坐标被标记为白色. 则图像中的黑点对应  $T[n]$  函数中低于平面  $g(x, y) = n$  的所有点<sup>[3]</sup>. 令  $C_n(M_i)$  表示集水盆地中点的坐标集合. 这个盆地与在第  $n$  阶段被水淹没的最小值有关, 即该集合中的灰度均小于  $M_i$ . 因此,  $C_n(M_i)$  也可看作下式所确定的一

收稿日期: 2008-10-08

基金项目: 江西省教育厅 2008 年科技计划项目 (GJJ08237); 华东交通大学校立科研资助项目 (08XX06)

作者简介: 张红斌 (1979-), 男, 江苏省如皋人, 讲师, 研究方向为图像处理、WEB 挖掘.

幅二值图像

$$C_n(M_i) = C(M_i) \cap T[n] \tag{2}$$

如果  $(x, y) \in T[n]$ , 则有  $C_n(M_i) = 1$ . 否则  $C_n(M_i) = 0$ . 所以, 只需在水溢出的第  $n$  个阶段使用“与”算子将  $T[n]$  中的二值图像分离出来即可得到该图像的分水岭.

令  $C[n]$  表示在第  $n$  个阶段集水盆地被水淹没部分的合集

$$C[n] = \bigcup_{i \in R} C_n(M_i) \tag{3}$$

令  $C[\max+1]$  表示所有集水盆地的合集

$$C[\max+1] = \bigcup_{i \in R} C(M_i) \tag{4}$$

可以看出处于  $C_n(M_i)$  和  $T[n]$  中的元素在算法执行期间是不会被替换的, 而这两个集合中的元素的数目与  $n$  保持同步增长. 因此,  $C[n-1]$  是集合  $C[n]$  的子集. 因此,  $C[n-1]$  中的每个连通分量也恰好是  $T[n]$  中的一个连通分量.

分水岭分割算法初始化时设定  $C[\min+1] = T[\min+1]$ . 然后算法进入递归调用, 假设在第  $n$  步, 已经构造出  $C[n-1]$ , 根据  $C[n-1]$  求  $C[n]$  的过程如下: 令  $Q$  代表  $T[n]$  中连通分量的集合, 对于每个连通分量  $q \in Q[n]$ , 存在下列 3 种可能性:

- (1)  $q \cap C[n-1]$  为空, 此时  $q$  是一个灰度为  $n-1$  的新的低洼;
- (2)  $q \cap C[n-1]$  包含  $C[n-1]$  的一个连通分量, 此时  $q$  位于某个集水盆地;
- (3)  $q \cap C[n-1]$  包含  $C[n-1]$  多于一个连通分量, 此时  $q$  包含多个集水盆地的分水线.

当遇到一个新的最小值符合上述条件 (1), 则将  $q$  并入  $C[n-1]$  构成  $C[n]$ . 当  $q$  位于某些局部最小值构成的集水盆地中时, 符合上述条件 (2), 此时也将  $q$  并入  $C[n-1]$  构成  $C[n]$ . 当遇到全部或部分分离成两个或更多集水盆地的山脊线时, 符合上述条件 (3), 此时若进一步注水会导致不同盆地的水聚在一起, 从而使水位趋于一致<sup>[4]</sup>. 因此, 必须建立一座水坝 (如果涉及到多个盆地就要建立多座水坝) 以阻止盆地内的水溢出<sup>[5]</sup>.

## 2 分水岭分割算法的实现

整个分水岭算法的运行流程如图 1 所示.

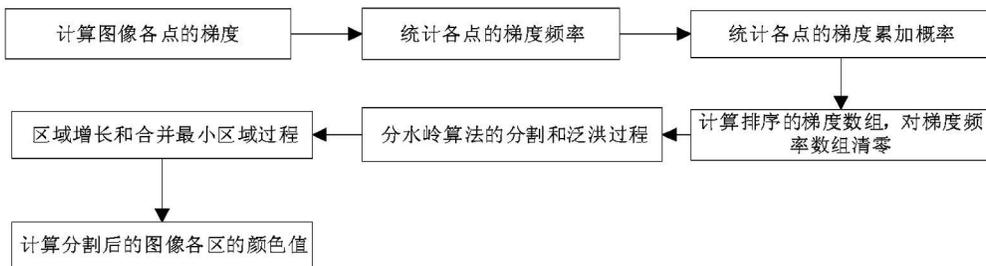


图 1 分水岭分割算法的流程图

下面给出 Inver 算法中实现分水岭分割和泛洪过程的伪代码描述, Vincent 算法不再赘述. 该过程是整个分水岭分割的关键, 在该过程中按顺序地处理已经排序好的梯度数组, 对图像中的每一像素点逐一查看其邻域点是否已被标识, 若已经标识, 则分配旧标识<sup>[6]</sup>, 否则, 分配一新的标识. 标识数组按行列顺序排列, 这样就能够很快根据行列位置找到图像中某一像素点的标识.

```

for (INT h = min h; h <= max h; h++) // 从最小灰度遍历到最大灰度
{
  INT stpos = gradientadd[h]; // 开始位置定义为当前最先灰度所对应的梯度
  INT edpos = gradientadd[h+1]; // 结束位置定义为比当前灰度大 1 所对应的梯度
  INT levelh ini = rgncount // 大于此编号者为本层新建 minima
  for (INT ini = stpos; ini < edpos; ini++)
  
```

```

{ NT x=graposarr[ini].x /获取当前像素点的 x坐标
NT y=graposarr[ini].y /获取当前像素点的 y坐标
NT cupos=y* imageWidth+x /获取当前像素点
myMath.GeNeilnt(x y cupos imageWidth imageHeight left right up down)
/找当前点的四邻域点,按左右上下顺序;
NT uppos=cupos-imageWidth; /获取当前点的上方邻接点
if((up>0) && (flag[uppos]>=0)) /处理上方邻接点
flag[cupos]=flag[uppos];
NT leftpos=cupos-1; /获取当前点的左方邻接点
if((left>0) && (flag[leftpos]>=0)) /处理左边邻接点
{ if(flag[cupos]>=0) /当前点存在标号
{ if(flag[cupos]==flag[leftpos])
{/此前已赋不同标号,要么有错误新区,要么该点为分水岭;
if(flag[cupos]>levelmini || flag[leftpos]>levelmini)
{/只要有一个大于 levelmini 则有错误新区;
if(flag[cupos]<=flag[leftpos])
{merge1arr[flag[leftpos]]=flag[cupos]; /记录合并;
flag[leftpos]=flag[cupos];
} else /否则为分水岭,不作处理;
{merge1arr[flag[cupos]]=flag[leftpos];
flag[cupos]=flag[leftpos]; }}}
} else flag[cupos]=flag[leftpos]; /此前未赋标号,给它加上标号; }
NT rightpos=cupos+1; /获取当前点的右方邻接点
if((right>0) && (flag[rightpos]>=0))
/代码类似处理左邻接点,故不再赘述;
NT downpos=cupos+imageWidth; /获取当前点的下方邻接点
if((down>0) && (flag[downpos]>=0))
/代码类似处理左邻接点,故不再赘述;
if(flag[cupos]<0) /说明该点邻域均未标识,可能为新的 minima
{ rgncount++;
/将标识号置为循环次数,这样标识号不会重复
flag[cupos]=rgncount
merge1arr[rgncount]=0; /合并数组中的值为 0 表示未被合并; }}}

```

### 3 分水岭分割实验结果分析

#### 3.1 粘连较少图像的分割结果

图 2-1~图 2-6是在选中一块图像区域后,分别采用松弛迭代分割、简单阈值分割、K均值聚类分割和分水岭分割后的分割结果.从图 2-1可以看出,该图像区域中细胞分化较好,即没有任何粘连现象,图中圆圈画出,共 6 个细胞.图 2-2简单阈值分割过程中,由于仅选择 1 个全局阈值分割,导致图中左下角的两个细胞在分割结果中出现了粘连,虚线圆圈标示,违背了原图中的细胞分布特性;图 2-3的 K 均值聚类分割, K 选择 3.由于图中细胞的灰度很接近,所以,在聚类过程中导致左下方的 4 个细胞完全粘连,同样用虚线圆圈标示,该分割也破坏了原图像中的细胞分布;图 2-4松弛迭代分割细胞的粘连更加严重,分割效果很差.而图 2-5和图 2-6的分水岭分割则可很好地把 6 个细胞分割开(实线标示),当然,图中也显示出一定的过分割现象,需要在分水岭分割的基础之上采用区域增长的方法合并多分割的小区域.



图 2-1 原图



图 2-2 简单阈值分割



图 2-3 K均值聚类分割



图 2-4 松弛迭代分割



图 2-5 Vincet分割



图 2-6 Inver分割

### 3.2 全局图像的分割对比

图 3-1~图 3-4是对一幅完整的胃上皮内肿瘤切片图像进行分割对比.从图 3-1中可以看出,原图像中包含三大图像子区域(红色虚线标示),其中,左上方图像子区域细胞之间有粘连,图像的对比度也很差,右上方图像子区域细胞之间粘连较少,且图像的对比如较左上方要好,而右下方图像子区域细胞之间几乎无任何粘连,且图像的对比如好,故这幅全局图像的“个性”比较严重,对其进行分割有一定的代表性.图 3-3是 Vincet分割结果,可见图中右下方的图像子区域分割良好,主要细胞均能被分割出来,但左上方和右上方两个图像子区域中,有粘连或对比如差的细胞就未能很好地分割出.图 3-4是 Inver分割的结果,可以看出,仅左上方图像子区域中的粘连细胞未很好分割出(在图 3-2中区域均值化后,该图像子区域中的细胞完全粘连),其它两个子区域均可对细胞进行较好的分割(右上方效果稍差,但可以获取部分细胞的分割结果),因此,可得出如下结论: Vincet分割对图像要求较高,即只有粘连较少且图像对比如较高的图像区域,该算法才能获取较好的分割效果.相比之下, Inver分割对图像的预处理结果则要求较低.

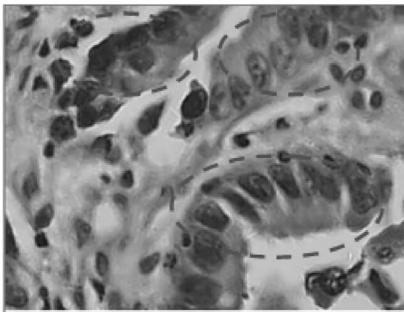


图 3-1 原图像

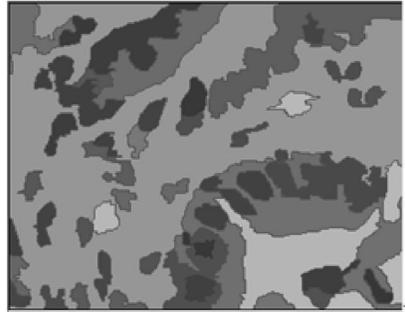


图 3-2 区域均值化图像



图 3-3 Vincet算法结果

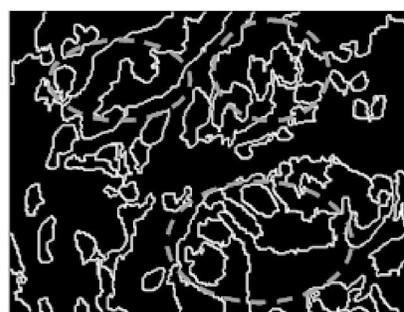


图 3-4 Inver算法结果

### 3.3 粘连严重图像的分割结果

图 4-1~图 4-6 显示了一幅粘连现象更加严重的细胞图像分别进行了简单阈值分割、K 均值分割、松弛迭代分割以及分水岭分割. 原图中共有 12 个细胞, 多个细胞之间有粘连, 绿色圆圈标示. 图 4-2 中简单阈值分割后上方细胞区域和右边细胞区域未能完全分离开, 左下方三个细胞之间还保留了极少的粘连, 分割后共得到 4 大目标区域, 分割效果很差. K 均值聚类分割效果也很差, 分割后粘连仍然严重, 右下方三个细胞保留了极少的粘连, 图中经过分割得到了 2 大目标区域. 迭代松弛分割结果稍好, 但上方和右方均未分割开, 分割后共得到 6 大目标区域. 此外, K 均值聚类和松弛迭代分割法容易把细胞内部误分割为背景像素 (图中黑色目标区域中存在大量与背景同色的像素). 图 4-5 和图 4-6 是分水岭分割, 可见, Inver 分割后共得到 7 大目标区域, 而 Vincent 分割后仅得 3 大目标区域, 但 Inver 的过分割稍多. 因此可得出如下结论: 对于粘连严重的目标图像, Inver 分割稍好.



图 4-1 原图



图 4-2 简单阈值分割结果

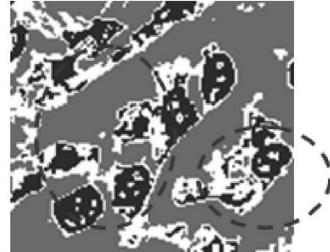


图 4-3 k 均值聚类分割

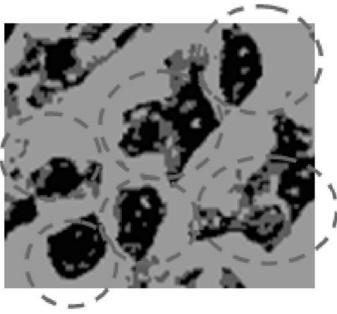


图 4-4 松弛迭代分割



图 4-5 Vincent 分割

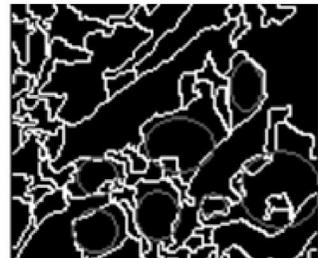


图 4-6 Inver 分割

### 3.4 实验数据分析

选择 100 组细胞样本 (粘连少的和粘连严重的各 50 组), 对本算法与 K 均值聚类等算法进行分割效果的对比实验, 表 1 列出所有能较好分割出的组数.

表 1 本算法与其它分割算法的比较

	松弛迭代法	K 均值聚类	Vincent 算法	Inver 算法
粘连少的 50 组	40 组	32 组	43 组	45 组
粘连多的 50 组	32 组	34 组	44 组	46 组

可以看出, 分水岭分割算法明显优于其它分割算法. 此外, 软件中还对分割出的图像进行了边界跟踪, 并计算边界跟踪之后的细胞区域的周长、面积、最小外接矩形的长宽比等, 其结果如表 2 和表 3 所示 (单位均是像素).

表 2 癌细胞图像特征数据记录 (5 组)

文件名	最小矩形 (长—宽)	平均长宽比	周长	平均周长	面积	平均面积
癌细胞 1	49—42		276		830	
癌细胞 2	40—32		126		741	
癌细胞 3	30—43	43—41	276	215.6	392	885.4
癌细胞 4	56—52		279		1 889	
癌细胞 5	40—37		121		575	

表 3 正常细胞图像特征数据记录 (5组)

文件名	最小矩形 (长—宽)	平均长宽比	周长	平均周长	面积	平均面积
正常细胞 1	20—15		60		169	
正常细胞 2	18—18		58		160	
正常细胞 3	18—17	19—16	53	6.4	187	170.4
正常细胞 4	18—16		57		161	
正常细胞 5	21—16		54		175	

表 2 中,癌细胞 4 的面积远大于其它图像,可能是因为图像中的癌细胞和其它细胞粘连在一起,分割后癌细胞和周围细胞连成一个大的闭合区域,从而使其面积数值明显区别于其它数据.而正常细胞粘连较少,所以没有出现某个细胞的形态参数远远大于其它细胞的现象.对比表 2 和表 3 中的 3 个平均数值(平均高度—宽度、平均周长、平均面积),可以看出,任何一个形态参数都具有可以把正常细胞和癌变细胞分离开的数值特性,如果合理地设计分类器,则可以完成图像中癌变细胞和正常细胞的自动分割.

## 4 结束语

给出了分水岭分割的基本流程,并实现了 Vincent 和 Inver 两种不同的分水岭分割算法.实验结果表明: Vincent 分割对图像的预处理要求较高,即只有粘连较少且对比度较高的图像区域,该算法才能获取较好的分割效果.相比之下, Inver 分割不但可以对粘连较少的图像区域进行合理分割,同时在粘连很严重的图像中,其分割效果也远胜于 Vincent 分割、阈值分割、K 均值分割以及松弛迭代分割等算法.笔者下一步将继续对更加优化的图像预处理算法和模式分类器的设计进行研究,此外本算法的时间复杂度过高是  $O(n^2)$ ,还有待进一步的改进.

## 参考文献:

- [1] 张红斌,甘 岚,等. 胃上皮肿瘤边界跟踪算法的实现 [J]. 计算机工程与应用, 2007, 43(4): 231—233.
- [2] 李广丽,刘遵雄,张红斌,等. 肿瘤图像病理诊断系统的设计 [J]. 华东交通大学学报, 2007, 24(5): 98—102.
- [3] 袁晓辉,许 东,夏良正,等. 基于形态学滤波和分水线算法的目标图像分割 [J]. 数据采集与处理, 2003, 18(4): 455—459.
- [4] PATNO. Fuzzy relations applied to minimize over segmentation in watershed algorithms [J]. Pattern Recognition Letters 2005, 26(6): 819—828.
- [5] 尹建军,毛罕平,王新忠,等. 不同生长状态下多目标番茄图像的自动分割方法 [J]. 农业工程学报, 2006, 22(10): 149—153.
- [6] 何东健. 数字图像处理 (第 2 版) [M]. 西安:西安电子科技大学出版社, 2008. 97—100.

## Application of Watershed Algorithm on Stomach Epidermis Tumor Segmentation

ZHANG Hongbin<sup>1</sup>, LI Guangli<sup>1</sup>, LIU Zunxiong<sup>1</sup>, HUANG Yizhen<sup>2</sup>

(1. School of Information Engineering East China Jiaotong University Nanchang 330013;

2. School of Software Beijing University Beijing 100871 China)

**Abstract** Image segmentation is a key process of image recognition. Relaxation iteration algorithm and K means cluster algorithm are used to resolve the stomach epidermis tumor segmentation, but some conglutinated cells cannot be separated by others. So the Vincent watershed algorithm and the Inver watershed algorithm are designed to conduct segmentation experiments about the stomach epidermis tumor. A lot of experiments about several kinds stomach epidermis tumor are done to build the segmentation theory. The conclusion of these experiments shows that these two algorithms could get good segmentation results when the cell image has little overlapping area. The Inver algorithm could get better result than the Vincent algorithm while the cell image has much more overlapping area. But Inver algorithm may have more useless segmentation areas than Vincent algorithm.

**Key words** stomach epidermis tumor; image segmentation; watershed algorithm; Vincent; Inver