

结合分水岭与自动种子区域生长的 彩色图像分割算法

杨家红 刘杰 钟坚成 何明志

(湖南师范大学电子与信息工程系, 长沙 410081)

摘要 为了对彩色图像进行有效分割, 提出了一种将分水岭算法与自动种子区域生长相结合的分割算法。该算法首先利用基于边缘信赖度的各向异性扩散方程对图像进行平滑预处理, 以便在去除图像噪声的同时, 保持图像的边缘信息, 从而有效解决了分水岭算法的过分割现象; 然后在此基础上, 设计了一种基于色调均值差的自动种子区域选取算法, 即根据色调均值和饱和度均值的相似性来对分水岭算法分割形成的区域进行种子区域生长, 并利用小图像区域消解算法对区域生长所遗留的小区域进行消解, 减少了图像中小区域的数目, 从而实现了彩色图像的有效分割。实验结果表明, 该算法简单、快速, 且能得到较其他算法更好的分割效果。

关键词 分水岭 自动种子区域生长 小图像区域消解 各向异性扩散 彩色图像

中图法分类号: TP391.41

文献标志码: A

文章编号: 1006-8961(2010)01-0063-06

A Color Image Segmentation Algorithm by Integrating Watershed with Automatic Seeded Region Growing

YANG Jia-hong LIU Jie ZHONG Jian-cheng HE Ming-zhi

(Department of Electronic & Information Engineering, Hunan Normal University, Changsha 410081)

Abstract A color image segmentation algorithm by integrating watershed with automatic seeded region growing is proposed in this paper. Anisotropic diffusion based on edge confidence is used for image smoothing preprocessing which can eliminate noise while maintain image edge information, and so that the watershed's over-segmentation problem is effectively solved. The algorithm of automatic seeded region selection based on the mean hue difference is designed to make seeded region growing based on the regions similarity of mean hue value and mean saturation value. Furthermore, an algorithm of small region removing is designed to removing the small region. As a result the color image segmentation is efficiently realized. The experimental results demonstrate that this algorithm is simple and fast, can obtain better segmentation results compared with other algorithms.

Keywords watershed, automatic seeded region growing, small region removing, anisotropic diffusion, color image

0 引言

图像分割就是将图像中感兴趣的部分划分出来的技术, 是模式识别和计算机视觉中的一个关键环节。由于彩色图像提供了丰富的色彩信息, 对人的

视觉感知极为重要, 因此对彩色图像进行分割的研究备受关注。在彩色模型系统中, 由于 HSV (色调、饱和度、数值) 彩色模型比红绿蓝 (RGB) 系统更接近于人们的经验和对彩色的感知, 因此本文算法采用了该彩色模型。目前有大量可用的彩色图像分割方法, 例如把图像分割问题看作是基于一颜色和空间

基金项目: 湖南省自然科学基金项目 (08JJ313); 国家烟草专卖总局科技开发项目 (2006371)

收稿日期: 2008-06-27 改回日期: 2008-11-06

第一作者简介: 杨家红 (1968—), 男, 教授。1997年获工学博士学位, 现为湖南大学控制科学与工程在站博士后。主要研究领域为模式识别与数字图像处理、网络信息安全、多媒体信息处理技术。E-mail: jhyang3668@vip.sina.com

特征的分类问题,可以分为有监督和无监督分类问题^[1]。另外,大部分的灰度图像分割技术也可以扩展到彩色图像,它们大致可以分为:边缘检测法、聚类方法、基于区域的方法 3 组。其中基于区域的方法包括区域生长、分水岭算法、分裂和归并,以及它们的混合技术。这些灰度图像分割算法不仅可以应用于不同的颜色空间,而且可以直接应用于每个颜色分量上^[2]。

本文采用基于区域的分割方法,将分水岭与区域生长、区域归并 3 种方法相结合,并根据分水岭算法对噪声图像较为敏感的特点,在分水岭处理之前,先采用改进的基于边缘信赖度的各向异性扩散算法对图像进行预处理^[3];然后经分水岭算法分割后,再进行区域生长将相互邻接的相似区域构造成一个区域,而对那些在生长中无法与邻接区域归并的小区域,则用小图像区域消解的方法将其归并到一个最相似的邻接区域中去。

1 分水岭初始分割

数学形态学^[4]的基础是作用于物体形状的非线性算子的代数,它在很多方面都要优于基于卷积的线性系统。与其他标准算法相比,在很多领域中,如预处理、基于物体形状的分割、物体量化,等等,数学形态学方法都能获得更好的处理结果和更快的速度。数学形态学图像处理理论中比较有代表性的算法是分水岭算法。

由于图像中噪声的存在,致使用分水岭算法分割图像容易产生过分割现象,文献[5]中提到了几种解决过分割的预处理技术,主要有中值滤波、彩色形态学、低通滤波和高斯滤波等。由于这些滤波技

术去除的是图像中频率较高的信息,因而对边缘信息的保持效果通常较差,Perona和Malik提出的各向异性的P-M扩散方程^[6]在去除噪声的同时,还能保持图像的边缘信息,本文采用了改进的基于边缘信赖度的各向异性扩散算法^[3]来对图像进行预处理。改进的基于边缘信赖度的各向异性扩散算法主要针对P-M扩散算法中 K 和 λ 的选取进行改进, K 是用于控制传导系统函数形状的参数, λ 是用来控制扩散程度的参数,边缘信赖度是用来衡量局部窗口内的图像像素与边缘相似程度的参数,边缘信赖度越大,该窗口的中心像素是边缘的可能性就越大。对于边缘信赖度大的像素,若减小其对应的 K 和 λ 值,则可以使边缘细节在扩散过程中得到保持;而对于边缘信赖度小的像素,若增大其对应的 K 和 λ 值,则能够加大其扩散程度和降低噪声。设图像 I 的像素 (x, y) 的边缘信赖度为 $O_{x, y}$,则本文采用如下局部扩散参数:

$$K_{x, y} = K e^{4(0.5 - O_{x, y})} \quad (1)$$

$$\lambda_{x, y} = \lambda e^{(0.5 - O_{x, y})/2} \quad (2)$$

此时改进的基于边缘信赖度的各向异性扩散方程为

$$I_{x, y}^{(l+1)} = I_{x, y}^{(l)} + \lambda_{x, y} (P_N \square_N I_{x, y} + P_S \square_S I_{x, y} + P_E \square_E I_{x, y} + P_W \square_W I_{x, y}) \quad (3)$$

其中,传导函数 P_{Dir} 是梯度 $\square_{Dir} I_{x, y}$ 的函数,下标 Dir 代表东(E)、南(S)、西(W)、北(N)4个计算梯度的方向,取 $P_{Dir} = e^{-(\square_{Dir} I_{x, y} / K_{x, y})^2}$,参数 $K_{x, y}$ 和 $\lambda_{x, y}$ 的取值在图像空间域是逐点变化的,即参数是针对边缘信赖度而局部自适应的。

图1就是经过改进的基于边缘信赖度的各向异性扩散算法扩散后的图像的分水岭算法分割结果。



(a) 原彩色图像



(b) 分水岭算法分割结果



(c) 经过改进的基于信赖度的各向异性扩散算法扩散后的图像分水岭算法的分割结果

图 1 改进的基于边缘信赖度的各向异性扩散对分水岭算法分割的影响

Fig. 1 Influence of watershed on an isotropic diffusion based on edge confidence

从图1可以看出,直接用分水岭算法分割产生了严重的过分割现象(见图1(b)),图中绿树部分

被分割成了大量的小区域。经过改进的基于边缘信赖度的各向异性扩散处理后的图像如图1(c)所

示, 由图 1(c) 可见, 分割后的区域个数相比图 1(b) 减少了近 50%, 相应也减小了计算量, 同时图像的轮廓也更加清晰。本文以下的种子区域选取、区域生长和小图像区域消解都是在此基础上进行的。

2 自动种子区域生长

经过分水岭算法分割后的图像含有大量的分割区域, 而如何从中选择正确的种子点则是本文种子区域生长算法的首要问题, 也是核心问题。文献 [7] 提出了将图像中的极小区域作为种子区域, 这里的极小区域指的是包含像素点数不到总图像像素点数的 1/400 的区域。而文献 [8] 则提出了一种自动的种子区域选取方法, 该方法主要依据以下两条规则来进行种子区域的选取: 1) 规则 1 首先对每个区域 R_i , 定义一个邻接区域集合 $V_R^i = R_i \cup \{R_j \mid R_j \in G, j=1, \dots, k\}$, 第 j 个区域 R_j 是第 i 个区域 R_i 的邻接区域; 然后通过相似性函数对区域 R_i 与集合 V_R^i 的所有区域的色调均值和饱和度均值进行相似性判断, 如果只有一个区域的相似性函数值大于一个阈值, 那么该区域才能被选择为种子区域; 2) 规则 2 要求一个区域与它的邻居区域的相对欧氏距离的最大值要小于一个阈值, 这里的欧氏距离的定义是采用区域的色调分量均值来进行计算。由于该自动种子区域选取方法需分别设置两个参数和两个阈值, 因此计算量大, 针对这些问题, 本文采用了一种更为简便的自动种子区域选取方法, 即基于色调均值差的自动选取算法, 因为色调是决定颜色本质的基本特性, 所以要求种子区域的色调均值与其邻接区域的色调均值差最大不能超过阈值。假如一个区域 R_i , 它的邻接区域是 $R_j (j=1, 2, \dots, k)$, k 是邻接区域的个数, 则它们的相似性计算如下:

$$D(t) = \frac{\|x(i) - x(t)\|}{x(i)} \quad t=1, 2, \dots, k \quad (4)$$

其中, $x(i)$ 是区域 R_i 的色调均值, $x(t)$ 是每个邻接区域的色调均值。

而区域 R_i 与它邻接区域的最大色调均值差为

$$D_{\max} = \max_{t=1}^k (D(t)) \quad (5)$$

满足以上相似性准则的区域就定义为种子区域, 这里的阈值可根据经验取为 0.03。如果取值过

低, 就会有更多的区域被选为种子区域, 而且不同的区域也会相互连接起来; 同样, 如果取值过高, 则所取得的种子区域数量就减少, 且图像中的某些对象就会失去^[9]。

区域生长就是对种子区域逐个去考查它与邻接区域的色调均值差及饱和度均值差是否都满足相似性判定阈值, 如果满足, 则将邻接区域标识成与种子区域相同的标记, 这样就构成一个区域。在进行区域生长之前, 对初始分割后的 N 个区域, 首先将各区域之间的邻接关系用一个 $N \times N$ 的邻接矩阵存储起来, 同时计算出每个区域的色调均值和饱和度均值; 然后进行自动种子区域选择, 将选取出的种子区域用一个数组标识。具体的生长算法如下:

算法 1 自动种子区域生长

输入: 经过改进的基于边缘信赖度的各向异性扩散的分水岭算法分割图像 J , 标识种子区域的数组 S , 邻接矩阵 N , 各区域的色调均值 x 和饱和度均值 y , 色调均值差阈值 T_1 , 饱和度均值差阈值 T_2 。

```

for  $S$  中的种子区域  $R_{seed}$ 
{
在  $N$  中找出种子区域  $R_{seed}$  的邻接区域集  $D$ 
if  $D \neq \emptyset$ 
{
for  $\forall R_{neighbor} \in D // D$  中的邻接区域  $R_{neighbor}$ 
{
*  $x_{SR}$  是种子区域的色调均值,  $y_{SR}$  是种子区域的饱和度均值,  $x_{NR}$  是邻接区域的色调均值,  $y_{NR}$  是邻接区域的饱和度均值 * /
if  $\|x_{SR} - x_{NR}\| \leq T_1 \cup \|y_{SR} - y_{NR}\| \leq T_2$ 
* 判断种子区域与其邻域的色调均值差和饱和度均值差是否都小于给定阈值 * /
{
 $J(i_{NR}, j_{NR}) \leftarrow SR$ 
* 把图像  $J$  中以  $NR$  为标记的元素标记为  $SR$  * /
 $N[SR, NR] \leftarrow 0$ 
 $N[NR, SR] \leftarrow 0$ 
// 在  $N$  中把  $R_{neighbor}$  和  $R_{seed}$  的邻接关系置 0
在  $N$  中找出  $R_{neighbor}$  的邻接区域集  $E$ 
for  $\forall R_{n-neighbor} \in E \cup R_{n-neighbor} \neq R_{seed}$ 
{
* 将  $E$  中除种子区域以外的区域的邻接关系重置 * /
 $N[SR, NNR] \leftarrow 1$ 
 $N[NNR, SR] \leftarrow 0$ 
 $N[NNR, NR] \leftarrow 0$ 
 $N[NR, NNR] \leftarrow 0$ 

```

* 在 N 中把 R_{seed} 和非邻接区域 $R_{n-neighbor}$ 的邻接关系置 1, $R_{neighbor}$ 和 $R_{n-neighbor}$ 的邻接关系置 0* /

输出: 区域生长后的图像, 区域生长后的邻接矩阵 N 。

根据经验, 阈值 T_1 的取值范围是 0.001 ~ 0.003 阈值 T_2 的取值是 0.001。

邻接矩阵 N 在生长过程中的变化可用图 2 的例子来描述, 图 2(a) 是一个 $n \times n$ 大小的邻接矩阵 N , 图中列出了前面一个 5×5 大小的矩阵块, B, C, D 表示其他矩阵块; 图 2(b) 表示当种子区域是标号为 3 的区域时, 它的邻接区域就有标号为 1, 4, 5 的区域, 根据相似准则可知, 假如它的相似区域是标号为 1, 5 的两个区域, 则经过生长后的局部邻接矩阵为 N 。

$$\begin{matrix}
 \begin{pmatrix} 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 1 & 0 \end{pmatrix} & \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix} \\
 \text{(a)} & \text{(b)}
 \end{matrix}$$

图 2 局部邻接矩阵在种子区域生长中的变化

Fig. 2 Transformation of partial adjacency matrix in seeded region growing

区域生长时, 如果种子区域选取方法不一样, 那么得到的生长效果也将不一样。如图 3 所示, 分别用两幅图来对文献 [7] 和文献 [8] 的种子区域选择方法与本文的方法所得到的区域生长结果进行比较。

从图 3 可以看出, 图 3(c) 的生长效果比图 3(a) 和图 3(b) 的生长效果要好, 图 3(f) 中用红色圈起的部分比图 3(d) 和图 3(e) 中用红色圈起的部分的轮廓细节分割得要好。文献 [7] 将极小区域作为种子区域, 且只考虑了区域的像素个数, 而生长则从极小的区域开始; 而本文的自动种子区域选择则是根据区域的颜色特征, 按照区域色调均值的相似准则进行选取, 由于可使得种子区域与其邻接区域中具有最高的相似性, 从而生长的结果更接近于所期望的区域。与文献 [8] 方法相比, 本文的种子选取方法简单、计算量小, 且分割效果好。

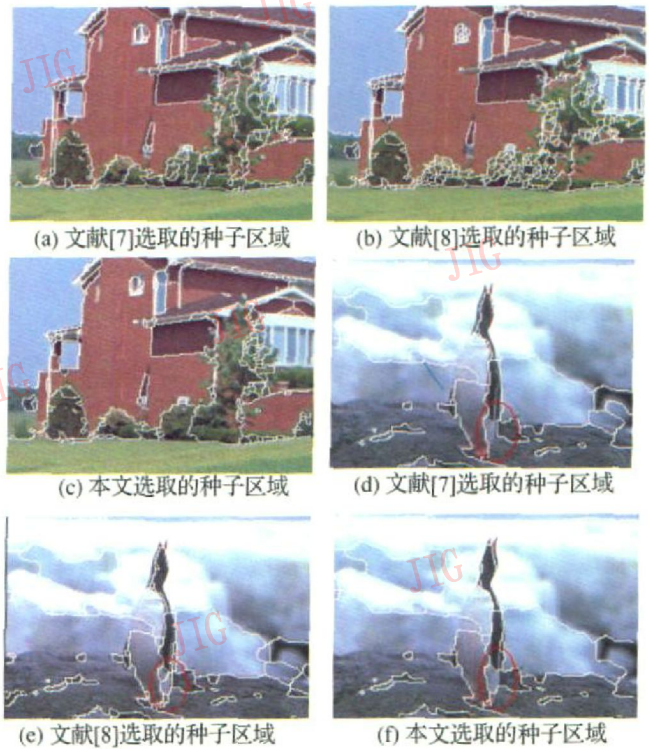


图 3 不同的种子区域选择方法的比较

Fig. 3 Comparison of different seeded region selection

3 小图像区域消解

使用区域生长方法分割后的图像, 由于参数设置的非最优性, 因而总会有一些小区域存在, 而这些小区域在进一步的分割中通常并不重要, 可以看作是分割噪声。为了减少分割后图像中小区域的数目, 可以通过设定一个阈值, 将所有区域尺寸小于这个阈值的区域归并到与它最相似的一个邻接区域中。

算法 2 小图像区域消解

算法描述如下:

输入: 包含 N 个区域 R_m 的区域生长后的图像、邻接关系矩阵 N , 各区域的面积 A 、各区域的色调均值 x 和饱和度均值 y 、阈值 T 。

```

for  $m = 1:N$ 
{
if  $A_m \leq T$  // 第  $m$  个区域的面积小于等于阈值
{

```

* x_m 为第 m 个区域 R_m 的色调均值, x_i 为邻接区域的色调均值, y_m 为第 m 个区域 R_m 的饱和度均值, y_i 为邻接区域的饱和度均值。 K 是邻接区域的个数 * /

$$S_i = \min(\sqrt{(x_m - x_i)^2 + (y_m - y_i)^2}), i = 1, 2, \dots, k$$

* 一致性准则的计算, 寻找与第 m 个区域 R_m 最相似

的一个邻接区域 R_i^* /

$$R_i = R_m \cup R_i$$

$$x_i \leftarrow (x_i \times A_i + x_m \times A_m) / (A_i + A_m)$$

$$y_i \leftarrow (y_i \times A_i + y_m \times A_m) / (A_i + A_m)$$

$$A_i \leftarrow A_i + A_m$$

/更新区域 R_i 的色调均值、饱和度均值、面积

}

}

输出: 小区域消除后的最终分割结果

算法中阈值的设定, 阈值主要依据图像中所要提取的对象的尺寸大小而定, 根据经验, 该阈值的取值范围为图像总像素数的 1/110 至 1/170。

图 4 分别给出了图 3 (c) 和图 3 (f) 经小图像区域消解后的最终分割结果。



(a) 图3(c)最终分割结果

(b) 图3(f)最终分割结果

图 4 小图像区域消解后的最终分割结果

Fig. 4 Final segmentation result after small region removing

4 实验及结果讨论

本文彩色图像分割算法步骤如下:

1) 将彩色图像从 RGB 彩色模型转换成 HSV 彩色模型;

2) 改进的基于边缘信赖度的各向异性扩散: 先将彩色图像转换成灰度图像, 再对灰度图像进行改进的基于边缘信赖度的各向异性扩散处理;

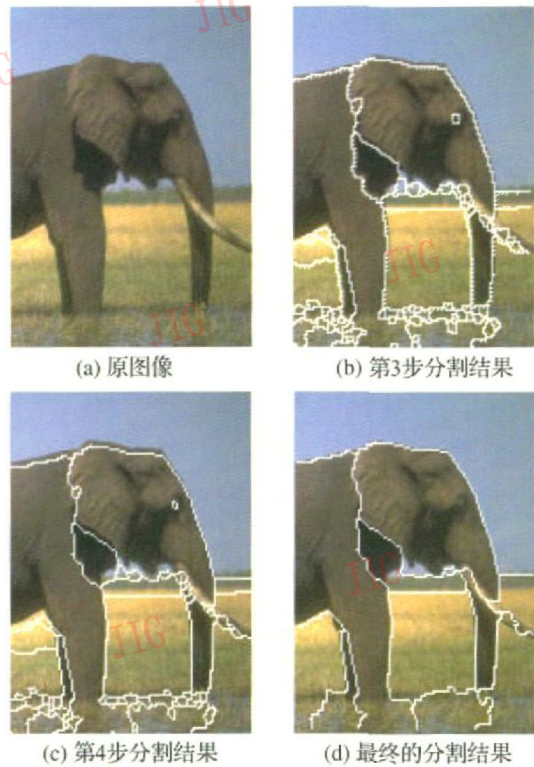
3) 分水岭算法分割: 先将扩散处理后的图像生成梯度图像, 再对梯度图像用分水岭算法进行分割^[10];

4) 自动种子区域生长: 先自动选取出种子区域, 然后根据自动种子区域生长算法, 将种子区域与其邻接区域中具有相同颜色特征的区域合并起来;

5) 小图像区域消解: 根据小图像区域消解算法, 将区域尺寸低于设定阈值的小图像区域合并到与其最相似的一个邻域中去。

图 5 的一个例子说明了本文算法的分割过程, 图 6 给出了本文算法的一些分割结果。

本文算法是将分水岭算法与自动种子区域生长



(a) 原图像

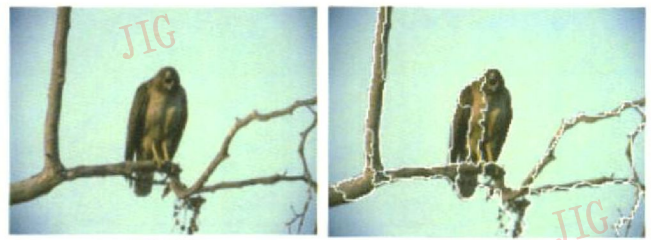
(b) 第3步分割结果

(c) 第4步分割结果

(d) 最终的分割结果

图 5 分割过程示意图

Fig. 5 Segmentation process sketch map



(a) 鹰原图

(b) 分割结果



(c) 象群原图

(d) 分割结果

图 6 本文算法的分割结果

Fig. 6 Segmentation result by the proposed algorithm

相结合。在分水岭分割前, 先用改进的基于边缘信赖度的各向异性扩散算法对图像进行预处理, 以便在增强图像边缘的同时, 又能去除图像的噪声, 从而改善了直接采用分水岭算法的分割效果, 并有效解

决了分水岭算法的过分割现象。本文的自动种子区域生长算法与传统的算法相比,它的主要特点有:

1)采用了基于区域的生长算法,而不是基于像素的生长算法,这是因为首先区域含有比像素更丰富的信息量,其次基于区域的生长比基于像素的生长在速度上也要快很多;2)在选择种子区域时,可按照区域色调均值的相似准则进行自动选取,使得选取出的种子在其邻接区域中更具有代表性,从而更加有利于区域的生长;3)利用了图的邻接矩阵来表示区域之间的邻接关系,借助邻接矩阵很容易判断两个区域是否相邻,并容易求得各个区域的度。在小图像区域消解过程中,在搜索最小的图像区域时,可直接将所有比预先选择的尺寸小的区域作为小图像区域,而不再需要依尺寸进行排序,这样就提高了运算性能。本文的小图像区域消解方法是基于区域归并的,但是归并算法并不是只针对两个相邻的区域,而是将一个区域与它所有的邻接区域相比较,然后找到最相似的那个邻接区域进行归并,这样就不需要对它设置相似性判断的阈值,也就避免了对每个邻接区域都要做一次相似判断的步骤。

5 结 论

本文提出了一种将分水岭算法与自动种子区域生长相结合的彩色图像分割算法。该算法先通过采用改进的基于边缘信赖度的各向异性扩散方程对图像进行预处理,解决了分水岭算法的过分割现象;然后利用分水岭算法分割形成的区域进行自动种子区域生长,而自动的种子区域的选取则根据相似性标准,使选取出的种子与其邻接区域具有最高的相似性,从而提高了区域生长的效果。为了减少区域生长后所遗留的小图像区域的数目,可根据小图像区域消解算法对生长后的图像进行后处理。实验结果表明,本文提出的彩色图像分割算法是可行、有效的,虽然使分水岭算法的过分割现象得到了改进,但如何更好地提高分水岭算法的分割效果,还需要进一步研究。

参考文献 (References)

- [1] Lin Kaiyan, Wu Jun-hui, Xu Lihong. A survey on color image segmentation techniques [J]. Journal of Image and Graphics, 2005, 10(1): 1-10. [林开颜, 吴军辉, 徐立鸿. 彩色图像分割方法综述 [J]. 中国图象图形学报, 2005, 10(1): 1-10.]
- [2] Yang C K, Tsai W H. Reduction of color space dimensionality by moment-preserving thresholding and its application for edge detection in color images [J]. Pattern Recognition Letters, 1996, 17(5): 481-490.
- [3] Lu Rong, Shen Yi. An ultrasound image segmentation scheme based on edge confidence and shape similarity [J]. Journal of Image and Graphics, 2008, 13(1): 69-74. [芦蓉, 沈毅. 基于边缘信赖度和形状相似性的超声图像分割方案 [J]. 中国图象图形学报, 2008, 13(1): 69-74.]
- [4] Sonka Milan, Hlavac Vaclav, Boyle Roger. Image Processing Analysis and Machine Vision (Third Edition) [M]. New York, USA: Thomson Learning, 2005: 235-237, 657.
- [5] Wojciech Bieniecki. Oversegmentation avoidance in watershed-based algorithms for color images [C] // Proceedings of the IEEE International Conference on Modern Problems of Radio Engineering, Telecommunication and Computer Science, Lviv, Ukraine: House of Lviv Polytechnic, 2004: 169-172.
- [6] Perona P, Malik J. Scale-space and edge detection using anisotropic diffusion [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 1990, 12(7): 629-639.
- [7] Wang Tong. Research of Color Image Segmentation Based on Mathematical Morphology [D]. Changchun: Jilin University, 2006. [王彤. 基于数学形态学的彩色图像分割算法研究 [D]. 长春: 吉林大学, 2006.]
- [8] Wang Jianan, Kong Jun. A region-based SRG algorithm for color image segmentation [C] // Proceedings of the IEEE Sixth International Conference on Machine Learning and Cybernetics, New York, USA: IEEE Computer Society, 2007: 1542-1547.
- [9] Shih Frank Y, Cheng Shou-xian. Automatic seeded region growing for color image segmentation [J]. Image and Vision Computing, 2005, 23(10): 877-886.
- [10] Yoo Jae Myeong, Dinh Goan Nguyen, Lee Gueesang. Segmentation by morphological reconstruction and non-linear diffusion [C] // Miyazaki T, Park Incheon, Wei Daming. Proceedings of the 7th IEEE International Conference on Computer and Information Technology, Los Alamitos, CA, USA: IEEE Computer Society, 2007: 701-708.