

## 融合多尺度分析和水平集的交互式图分割方法

汪云飞<sup>\*①</sup> 毕笃彦<sup>①</sup> 史德琴<sup>①</sup> 黄天军<sup>②</sup> 刘迪<sup>①</sup>

<sup>①</sup>(空军工程大学工程学院 西安 710038)

<sup>②</sup>(93968部队 乌鲁木齐 830075)

**摘要:** 针对快速水平集算法用于图像分割时,存在水平集初始化和阈值设置的困难,该文提出一种融合金字塔模型、随机游走及水平集(PYR-RW-LS)的新算法。首先将多尺度分析引入随机游走算法,把分割结果作为快速水平集算法的初始化曲线,解决其初始化问题;接着把水平集演化看成对曲线上的点不断进行模式分类的过程,引入贝叶斯分类决策和最小距离分类决策交替工作,产生曲线演化所需的驱动力,同时将两种分类决策的失效条件作为新算法迭代停止的条件,解决了快速水平集算法阈值设置的困难。仿真实验结果表明:PYR-RW-LS算法比只采用模式分类思想的快速水平集算法拥有更高的计算效率,且在抗噪性方面亦优于随机游走算法,同时保留了随机游走算法对弱边缘不敏感的优点,尤其适用于大尺寸,高清晰度的图像处理。

**关键词:** 图像分割; 多尺度分析; 图论; 随机游走; 快速水平集

中图分类号: TP391

文献标识码: A

文章编号: 1009-5896(2012)09-2078-07

DOI: 10.3724/SP.J.1146.2012.00005

## An Interactive Graph Partition Method Through Combination of Multi-scale Analysis and Level Set

Wang Yun-fei<sup>①</sup> Bi Du-yan<sup>①</sup> Shi De-qin<sup>①</sup> Huang Tian-jun<sup>②</sup> Liu Di<sup>①</sup>

<sup>①</sup>(Engineering College of Air Force Engineering University, Xi'an 710038, China)

<sup>②</sup>(Troop 93968, Urumqi 830075, China)

**Abstract:** In the application of image segmentation based on fast level set algorithm, there exist difficulties in level set initialization and setting thresholds, so a new algorithm which combining PYRamid model, Random Walk and Level Set (PYR-RW-LS) is proposed. First, the multi-scale analysis technique is introduced into Random Walk (RW) algorithm, and its partition result is taken as the initialized curve of the fast level set algorithm, so the fast level set algorithm's initialization problem is solved; Then the evolution of the level set can be seen as the constant pattern classification of the points on the curve. Both Bayesian classification rule and minimal distance classification rule were introduced by this new algorithm to work alternatively, in order to acquire the driving force for curve evolution. And the invalidation conditions for both of the classification rules are set as the iteration stop conditions in this new algorithm, thus solving the difficulties in setting thresholds. Simulating experimental results show that PYR-RW-LS not only runs faster than the fast level set algorithm, which only adopts pattern classification ideas, but also has better capabilities than RW algorithm in terms of anti-noise capabilities; And the advantages of being insensitive to blurry boundaries remains with the RW algorithm. PYR-RW-LS algorithm, therefore, is good in particular, for images with large size and high resolution.

**Key words:** Image segmentation; Multi-scale analysis; Graph theory; Random Walk (RW); Fast Level Set (LS)

### 1 引言

在许多图像分析问题中,如何精确提取 $\Omega_0/\Omega_b$ (目标/背景)一直是一个重要且具有挑战性的课题。按 $\Omega_0$ 提取方式的不同可分为自动/半自动两类分割方法<sup>[1,2]</sup>,半自动分割方法按引导方式不同可分为以

下几类:(1)预先指定 $\Omega_0$ 的大致边界通过曲线演化完成最终分割;(2)预先标识 $\Omega_0$ 的部分边界,用最短路径方法拼接起来;(3)预先在 $\Omega_0$ 内标记一些种子点,以种子点提供的信息进行分割<sup>[2]</sup>。随机游走(RW)作为一种采用多标记点引导的交互式图分割方法<sup>[3]</sup>,将问题转化为线性方程系统的求解,稳定性得到了提高,但RW算法的计算效率和方程的求解规模相关<sup>[4]</sup>。随着现代数字照相技术的不断发展,图片尺寸和清晰度均在不断增大和提高,这必然导致RW算

2012-01-05 收到, 2012-04-17 改回

国家自然科学基金(61175029)资助课题

\*通信作者: 汪云飞 wdcpost\_2006@yahoo.com.cn

法的实时处理能力不断下降,很多学者在其基础上不断改进<sup>[5,6]</sup>。本文将RW算法与多尺度分析方法相结合,提出了采用金字塔模型的随机游走(PYR-RW)算法,弥补了RW算法在处理大尺寸图像时速度和抗噪性方面的不足;又引入模式分类思想,提出了改进的Shi算法,解决了快速水平集算法阈值设置的困难;综合两种算法的优点提出一种新的交互式图分割(PYR-RW-LS)算法,该算法在保证分割精度的前提下,仍可实现快速、稳定的鲁棒性分割。

## 2 PYR-RW-LS 算法

### 2.1 PYR-RW 算法

本文将多尺度分析<sup>[7]</sup>引入RW算法提出了采用金字塔模型的随机游走(PYR-RW)算法,该算法首先将图像进行粗化处理,得到图像序列 $I^k(k=1, \dots, m)$ ,接着将每层图节点分别建立4连接的图拓扑结构<sup>[8]</sup>,各层间通过标准的4叉树结构连接,构造出多分辨率金字塔,并在顶层图中应用RW算法,将提取的目标 $\Omega_0$ 投影到底层图中得到最终分割结果。本文构造的图金字塔结构如图1所示。

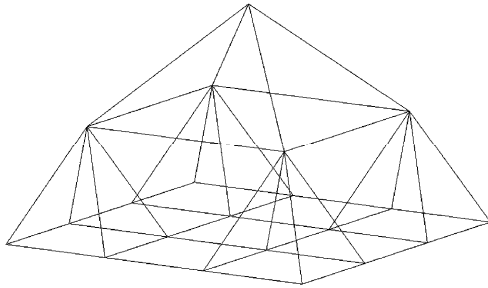


图1 图金字塔结构

### 2.2 基于模式分类的Shi算法

文献[9]提出的快速水平集算法,在本质上属于窄带算法的一种离散化表述形式<sup>[9,10]</sup>。该算法存在一个明显缺陷:即需要人工设置专门的阈值门限作为驱动力函数 $h(\mathbf{x})$ ,这在实际中很难把握。针对上述缺陷,引入模式分类思想将 $\Omega_0/\Omega_b$ 的提取过程抽象为一个二分类问题,则曲线演化过程可看成是对曲线上的点 $\mathbf{x}$ 不断进行模式分类的过程,假设处理的是灰度图像,本文提出的驱动力函数 $h(\mathbf{x})$ 形式为

$$\begin{aligned} h(\mathbf{x}) &= f(\mathbf{x}) + g(\mathbf{x}) = \lg \frac{\varepsilon + p(\Omega_0) \cdot p(I(\mathbf{x}) | \Omega_0)}{\varepsilon + p(\Omega_b) \cdot p(I(\mathbf{x}) | \Omega_b)} \\ &\quad + \lg \frac{\varepsilon + (I(\mathbf{x}) - c_b)^2}{\varepsilon + (I(\mathbf{x}) - c_o)^2} \\ &= \lg \frac{\varepsilon + p(\Omega_0) \cdot p(I(\mathbf{x}) | \Omega_0) \cdot (I(\mathbf{x}) - c_b)^2}{\varepsilon + p(\Omega_b) \cdot p(I(\mathbf{x}) | \Omega_b) \cdot (I(\mathbf{x}) - c_o)^2} \quad (1) \end{aligned}$$

其中 $p(\Omega_0)$ ,  $p(I(\mathbf{x}) | \Omega_0)$ 分别为目标类的先验概率和

类条件概率密度; $p(\Omega_b)$ ,  $p(I(\mathbf{x}) | \Omega_b)$ 分别为背景类的先验概率和类条件概率密度; $c_o$ 和 $c_b$ 分别指代 $\Omega_0$ 和 $\Omega_b$ 的灰度均值;特征向量 $I(\mathbf{x})$ 为点 $\mathbf{x}$ 处的灰度值; $\varepsilon$ 是一个极小量防止 $h(\mathbf{x})$ 计算时出现被零除的情况。且 $f(\mathbf{x})$ 是基于最小错误率时贝叶斯决策规则对应的分类器;而 $g(\mathbf{x})$ 是基于方差的最小距离分类决策对应的分类器;两种分类器交替工作产生曲线演化的驱动力,当 $h(\mathbf{x})$ 满足式(2)的所有判断条件时,两种分类决策同时失效,此时 $f(\mathbf{x}) = g(\mathbf{x}) = h(\mathbf{x}) = 0$ ,  $h(\mathbf{x})$ 不再产生曲线演化的驱动力,算法迭代停止。

$$\left. \begin{aligned} p(\Omega_0) \cdot p(I(\mathbf{x}) | \Omega_0) &= p(\Omega_b) \cdot p(I(\mathbf{x}) | \Omega_b) \\ (I(\mathbf{x}) - c_o)^2 &= (I(\mathbf{x}) - c_b)^2 \end{aligned} \right\} \quad (2)$$

假设采用该 $h(\mathbf{x})$ 的快速水平集算法为基于模式分类的文献[9]算法,该算法可以自适应的根据图像全局信息控制水平集曲线自动演化,避免了阈值设置的困难。该算法仍存在一些不足:体现在水平集的初始化位置对分割结果影响较大,不仅要求初始化时水平集曲线必须包含一定的 $\Omega_0$ 信息,而且存在 $\Omega_0$ 轮廓提取不唯一的问题。

### 2.3 PYR-RW-LS 算法的提出

PYR-RW算法随着金字塔层数的增多易产生无法精确提取 $\Omega_0$ 轮廓的问题,该问题本质上是由于高层图金字塔大量图像信息的损失造成的。考虑到基于模式分类的文献[9]算法通过曲线演化可以弥补PYR-RW算法精度不足的缺陷,而PYR-RW算法提取的 $\Omega_0$ 轮廓又可以解决基于模式分类的Shi算法存在的初始化问题,本文将两种方法结合,提出了一种新的PYR-RW-LS算法,该算法由PYR-RW部分和LS部分组成,实现细节如下:

在PYR-RW部分中图金字塔的构建属于线下处理阶段<sup>[7]</sup>,相当于粗化处理;采用如图2所示的映射规则,即 $\tilde{I}\mathbf{x} = \sum I\mathbf{x}/4 = \sum_{i=1}^4 I_i/4$ ,这样图像从第 $k$ 层映射到第 $k+1$ 层时粗化率为1/4;其余均为线上处理阶段<sup>[7]</sup>,相当于粗略分割;假设图金字塔结构共有 $m$ 层,令 $\Omega_{o/k}$ 和 $\Omega_{b/k}$ ( $k=2, \dots, m$ )分别代表第 $k$ 层图像的目标区域和背景区域, $\Omega_k$ ( $k=2, \dots, m$ )代表第 $k$ 层图像的全部区域,在顶层图 $G^m$ 中令 $\sum(\mathbf{x} \in \Omega_i)$ 代表区域 $i$ 中所有像素点的个数总和,则先验概率 $p(\Omega_{o/m})$ 可近似表示为 $p(\Omega_{o/m}) = \sum(\mathbf{x} \in \Omega_{o/m}) / \sum(\mathbf{x} \in \Omega_m)$ ,  $p(\Omega_{b/m})$ 可近似表示为: $p(\Omega_{b/m}) = \sum(\mathbf{x} \in \Omega_{b/m}) / \sum(\mathbf{x} \in \Omega_m)$ 。假设第 $k$ 层有 $r_k \times c_k$ 个元素,映射到第 $k+n$ 层时约有 $r_k \times c_k \times 1/4^n$

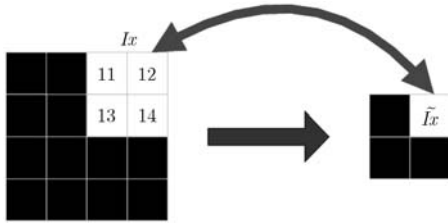


图2 层间映射准则

个元素，则  $p(\Omega_{o/k+n}) = (1/4^n \times \sum(\mathbf{x} \in \Omega_{o/k})) / |1/4^n \times \sum(\mathbf{x} \in \Omega_k)| = p(\Omega_{o/k})$ ，同理得到  $p(\Omega_{b/k+n}) = p(\Omega_{b/k})$ ，即金字塔顶层计算出的  $p(\Omega_{o/m})$  和  $p(\Omega_{b/m})$  和底层原始图像的  $p(\Omega_o)$  和  $p(\Omega_b)$  相等。LS 部分主要依靠式(1)的驱动力函数  $h(\mathbf{x})$  产生曲线演化的动力，相当于精细分割<sup>[7]</sup>。因为  $\Omega_o$  和  $\Omega_b$  并不一定服从高斯分布，所以用  $\Omega_o$  或  $\Omega_b$  内不同像素点个数的统计直方图来近似模拟类条件概率密度  $p(I(\mathbf{x})|\Omega_o)$  和  $p(I(\mathbf{x})|\Omega_b)$ ，令  $\text{Num}(I(\mathbf{x}))(i=o,b)$  表示区域  $\Omega_o/\Omega_b$  中灰度值和  $\mathbf{x}$  处相同的像素点个数总和， $\sum_{\mathbf{x} \in \Omega_o} I(\mathbf{x})(i=o,b)$  表示区域  $\Omega_o/\Omega_b$  中和  $\mathbf{x}$  处相同的像素点灰度值总和，则  $p(I(\mathbf{x})|\Omega_o)$ ， $p(I(\mathbf{x})|\Omega_b)$  及均值  $c_o$ ， $c_b$  的表达式为

$$\left. \begin{aligned} p(I(\mathbf{x})|\Omega_o) &= \text{Num}(I(\mathbf{x})) / (\sum(\mathbf{x} \in \Omega_{o/m}) \times 4^{m-1}) \\ c_o &= \sum_{\mathbf{x} \in \Omega_o} I(\mathbf{x}) / (\sum(\mathbf{x} \in \Omega_{o/m}) \times 4^{m-1}) \\ p(I(\mathbf{x})|\Omega_b) &= \text{Num}(I(\mathbf{x})) / (\sum(\mathbf{x} \in \Omega_{b/m}) \times 4^{m-1}) \\ c_b &= \sum_{\mathbf{x} \in \Omega_b} I(\mathbf{x}) / (\sum(\mathbf{x} \in \Omega_{b/m}) \times 4^{m-1}) \end{aligned} \right\} (3)$$

为了提高驱动力函数  $h(\mathbf{x})$  的抗噪性，本文选取高斯模板  $G(\mathbf{x})$  为滤波窗口对  $h(\mathbf{x})$  进行滤波处理，如图3所示，用卷积的形式将滤波表示为

$$h'(\mathbf{x}) = G(\mathbf{x}) \otimes h(\mathbf{x}) \quad (4)$$

本文提出的PYR-RW-LS算法流程如表1所示。

1/16	1/8	1/16
1/8	1/4	1/8
1/16	1/8	1/16

图3 高斯模板

表1 PYR-RW-LS 算法流程

初始化：假设  $\Omega_o$  和  $\Omega_b$  分别代表原始图像  $I^1$  的目标区和背景区， $\Omega_{o/k}$  和  $\Omega_{b/k}$  ( $k=2, \dots, m$ ) 分别代表第  $k$  层图像的目标区和背景区，且  $L_{out}$  代表水平集演化的外部链表， $L_{in}$  代表水平集演化的内部链表， $F_{ext}(\mathbf{x})$  表示曲线在点  $\mathbf{x}$  处的演化方向， $N_4(\mathbf{x})$  代表点  $\mathbf{x}$  的4邻域，Nout 和 Nin 分别为水平集算法中两个内循环的迭代次数。

(1)线化处理 将原始图像  $I^1$  按照示意图2的原理进行粗化处理，生成图像序列  $I^k$  ( $k=1, \dots, m$ )，图像序列间通过标准的4叉树连接，构建如图1所示的图拓朴结构。

(2)线上处理

步骤1 种子点映射：在  $I^1$  上用种子点标定  $\Omega_o$  和  $\Omega_b$ ，并将种子点按照构造的图拓朴结构映射到  $I^m$  中，标定该层图像的  $\Omega_{o/m}$  和  $\Omega_{b/m}$ 。

步骤2 粗略分割：根据映射到  $I^m$  中种子点信息，应用 RW 算法，提取出  $\Omega_{o/m}$  轮廓。同时计算出顶层图的  $p(\Omega_{o/m})$  和  $p(\Omega_{b/m})$ ，得到  $I^1$  层的先验概率  $p(\Omega_o)$  和  $p(\Omega_b)$ 。

步骤3 分割结果投影： $\Omega_{o/m}$  轮廓按照图1的图拓朴结构重新投影到  $I^1$  中，作为水平集的  $L_{in}$  链表；同时生成  $L_{out}$  链表，且由式(3)计算出  $I^1$  层的类条件概率密度  $p(I(\mathbf{x})|\Omega_o)$ ， $p(I(\mathbf{x})|\Omega_b)$ ，均值  $c_o$  和  $c_b$ 。

步骤4  $I^1$  层上水平集的曲线演化：

For  $i=1$ : Nout do

(a)曲线朝外演化：遍历链表  $L_{out}$ ，根据式(1)计算  $L_{out}$  中每一点  $\mathbf{x}$  的  $h(\mathbf{x})$ ，若  $h(\mathbf{x}) > 0$ ，设置  $F_{ext}(\mathbf{x}) = 1$ ，即曲线朝外演化， $\forall \mathbf{x} \in L_{in}$ ，如果  $\exists \mathbf{y} \in N_4(\mathbf{x})$  满足  $\mathbf{y} \in \Omega_b$ ，设置  $\mathbf{y} \in L_{out}$ 。

(b)  $L_{in}$  冗余点消除：遍历链表  $L_{in}$ ，对于每个像素点  $\mathbf{x} \in L_{in}$ ， $\forall \mathbf{y} \in N_4(\mathbf{x})$  若满足  $\mathbf{y} \in L_{in}$  或  $\mathbf{y} \in \Omega_o$ ，则将  $\mathbf{x}$  从  $L_{in}$  中删除，并设置  $\mathbf{x} \in \Omega_o$ 。

(c)  $h(\mathbf{x})$  组件更新：更新先验概率  $p(\Omega_o)$ ， $p(\Omega_b)$ ；条件概率密度  $p(I(\mathbf{x})|\Omega_o)$ ， $p(I(\mathbf{x})|\Omega_b)$ ；目标区和背景区内均值  $c_o$  和  $c_b$ 。

(d)曲线朝内演化：遍历链表  $L_{in}$ ，根据式(1)计算  $L_{in}$  中每一点  $\mathbf{x}$  的  $h(\mathbf{x})$ ，若  $h(\mathbf{x}) < 0$ ，设置  $F_{ext}(\mathbf{x}) = -1$ ，即曲线朝内演化， $\forall \mathbf{x} \in L_{out}$ ，如果  $\exists \mathbf{y} \in N_4(\mathbf{x})$  满足  $\mathbf{y} \in \Omega_o$ ，设置  $\mathbf{y} \in L_{in}$ 。

(e)  $L_{out}$  冗余点消除：遍历链表  $L_{out}$ ，对于每个像素点  $\mathbf{x} \in L_{out}$ ， $\forall \mathbf{y} \in N_4(\mathbf{x})$  若满足  $\mathbf{y} \in L_{out}$  或  $\mathbf{y} \in \Omega_b$ ，则将  $\mathbf{x}$  从  $L_{out}$  中删除，并设置  $\mathbf{x} \in \Omega_b$ 。

(f)  $h(\mathbf{x})$  组件更新：更新先验概率  $p(\Omega_o)$ ， $p(\Omega_b)$ ；条件概率密度  $p(I(\mathbf{x})|\Omega_o)$ ， $p(I(\mathbf{x})|\Omega_b)$ ；目标区和背景区内均值  $c_o$ ， $c_b$ 。

(g)停止条件判断： $\forall \mathbf{x} \in L_{out}$  或  $\forall \mathbf{x} \in L_{in}$  若同时满足式(2)的判断条件，跳至步骤5，否则返回步骤4 继续执行。

步骤5  $I^1$  层上演化曲线的平滑处理：

For  $j=1$ : Nin do

演化步骤与步骤4 基本相同，只需采用式(4)的滤波器和图3所示的高斯模板对曲线进行平滑处理，并忽略式(2)的停止判断条件。

步骤6 水平集演化主循环停止条件判断：

$\forall \mathbf{x} \in L_{out}$  或  $\forall \mathbf{x} \in L_{in}$  若同时满足式(2)的判断条件，则结束曲线演化，算法停止，否则转到步骤4 重新执行。

### 3 实验结果及分析

实验用不同颜色的种子点分别标定  $\Omega_o$  和  $\Omega_b$ ，且种子点个数限定在 300 以内，PYR-RW 部分在运算时暂不考虑线下处理所损耗的时间；LS 部分将水平集的迭代次数上限设置为 3000 次。本文采用图像的全局方差衡量图像的清晰程度<sup>[11]</sup>，从表 3 的实验数据可以看出随着图金字塔层数的递增，方差呈下降趋势，即图像越来越模糊。

#### 3.1 图金字塔结构的有效性

图 4 中针对较大尺寸的 chest 图像，构建 4 层图金字塔结构。图 4(a), 4(c), 4(f), 4(i) 分别为构建的图金字塔结构中  $I^1, I^2, I^3, I^4$  层效果图，以及标记种子点在  $I^1, I^2, I^3, I^4$  层上的映射效果，从实验可以看出经过粗化处理  $I^1, I^2, I^3, I^4$  层图像尺寸得到显著降低，因为每次映射时  $I^{k+1}$  比  $I^k$  层会减少  $3/4$  的像素点，所以图像的清晰度也在不断下降。图 4(b) 为  $I^1$  层上用 RW 方法提取的  $\Omega_o$  轮廓；图 4(d), 4(g), 4(j) 分别为 PYR-RW 部分在  $I^2, I^3, I^4$  层提取的  $\Omega_o$  轮廓，而图 4(e), 4(h), 4(k) 分别为经 PYR-RW 部分粗略分割后投影到  $I^1$  层的  $\Omega_o$  轮廓；从图 4 可以看出伴随图像尺寸的缩小会损失大量的图像信息，导致 PYR-RW 部分在提取  $\Omega_o$  初始轮廓时精度也呈下降趋势，如图 4(d), 4(g), 4(j) 所示，造成的直接影响是 PYR-RW-LS 算法的 LS 部分在处理时迭代次数的增多和迭代时间的延长，可从表 2 的实验数据看出。图 5(a), 5(b) 分别给出了 chest 图像在采用 2 层图金字塔结构时，经 PYR-RW-LS 算法的 PYR-RW 部分粗分割后  $\Omega_o$  和  $\Omega_b$  的初始分类；及 LS 部分对 PYR-RW 部分得到的  $\Omega_o$  轮廓进行后续处理，分类完成时的  $p(\mathbf{x} | \Omega_o)$  分布曲线图，其中  $p(\mathbf{x} | \Omega_o)$  采用 64 bin 的灰度直方图来描述，通过统计  $\Omega_o / \Omega_b$  中不同像素值的个数来实现。从图 5 中可看出经过 LS 的后期处理， $\Omega_o$  和  $\Omega_b$  最终能达到趋于理想的分类效果。

#### 3.2 抗噪性测试

乘性噪声的抗噪性测试：通过将图像中每个像素点乘以一个单位均值的高斯变量产生，同时伴有 1-20% 的光照变化，对 Human 图像添加大小在 [0 1] 之间的乘性噪声，信噪比<sup>[11]</sup>的变化范围在 [3.973, 31.122] dB 之间。图 6(a), 6(b) 分别为原始图像，手动标记的  $\Omega_o$  轮廓；图 6(c) 为信噪比为 3.973 dB 时，对 Human 图像的  $\Omega_o / \Omega_b$  种子点标记效果；而图 6(d) - 6(e) 分别为 RW 和 PYR-RW-LS 算法提取的  $\Omega_o$  轮廓；图 6(f) 为本文算法的分割结果叠加到原图的效果。从图 6 可以看出添加最强的乘性噪声时，RW 算法已经无法提取出  $\Omega_o$  的轮廓，而 PYR-RW-LS

算法仍可以提取出  $\Omega_o$  的准确轮廓。图 7 为乘性噪声的抗噪性实验结果。

模糊噪声的抗噪性测试：采用尺寸为  $20 \times 20$ ，且标准差  $\delta$  的变化范围为 [1, 7] 的高斯低通滤波作为模糊核，逐步对 Blood 图像进行模糊化处理，信噪比的变化范围为 [5.514, 30.545] dB。图 8(a), 8(b) 分别为原始图像，手动标记的  $\Omega_o$  轮廓；图 8(c) 为信噪比为 5.514 dB 时，对图像的  $\Omega_o / \Omega_b$  种子点标记效果；而图 8(d), 8(e) 分别为 RW 和 PYR-RW-LS 算法提取的  $\Omega_o$  轮廓；图 8(f) 为本文算法的分割结果叠加到原图的效果。从图 8 可以看出当信噪比为 5.514 dB 时，RW 算法已经失效，而 PYR-RW-LS 算法仍可较准确的提取出  $\Omega_o$  的轮廓，取得较好的分割效果。从图 7 和图 9 可以看出随着乘性噪声或模糊噪声的信噪比不断减少，PYR-RW-LS 算法始终能保持对  $\Omega_o$  的高检测率，低虚警率和  $\Omega_o / \Omega_b$  的高正确分割率<sup>[12]</sup>。而 RW 算法随着信噪比的不断减少，对  $\Omega_o$  的检测率始终呈不断下降趋势，后期检测率的大幅上升是由于分割时引入了虚警率所导致，且 RW 算法对  $\Omega_o / \Omega_b$  的正确分割率始终呈下降趋势。综上可知本文所提算法对两种噪声的抑制性均强于 RW 算法<sup>[3]</sup>。

#### 3.3 算法速度和弱边缘测试

从表 2 的实验数据可以看出，PYR-RW-LS 算法在运算速度上都要明显强于采用模式分类的文献 [9] 算法，且 PYR-RW 部分和 LS 部分在处理图像时运算时间和迭代量也得到大幅降低。对于图 10 所示的弱边缘图像，图 10(a) 为少数标记点对  $\Omega_o / \Omega_b$  的标记效果；图 10(b) 为 PYR-RW 部分得到的  $\Omega_o / \Omega_b$  概率分布；图 10(c), 10(d) 分别为 RW 算法及 PYR-RW-LS 算法的 PYR-RW 部分提取的  $\Omega_o$  轮廓。可以看出 PYR-RW-LS 算法对弱边缘也不敏感。

### 4 结束语

本文将多尺度分析方法引入 RW 算法，并且与基于模式分类思想的文献 [9] 算法相结合，提出了一种新的 PYR-RW-LS 算法，该算法比 RW 算法拥有更强的抗噪性能，且在时间效率上也要高于只采用模式分类思想的文献 [9] 算法，并保留了 RW 算法对弱边缘不敏感的特性，尤其适用于大尺寸，高清晰度的图像处理。下一步的工作将集中在 (1) 如何根据不同尺寸的图像选择合适的金字塔层数；(2) 测试种子点数目对算法分割结果的影响；(3) 针对不同的图像类型如何选用不同的统计直方图来模拟式 (1) 中的  $p(I(\mathbf{x}) | \Omega_o)$ <sup>[11, 12]</sup>，进一步扩展 PYR-RW-LS 算法的适用范围。



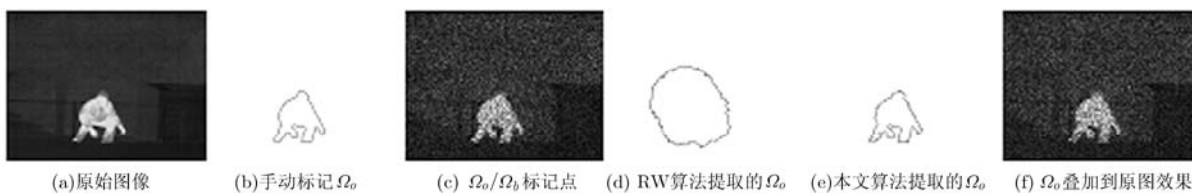


图 6 乘性噪声的实验结果

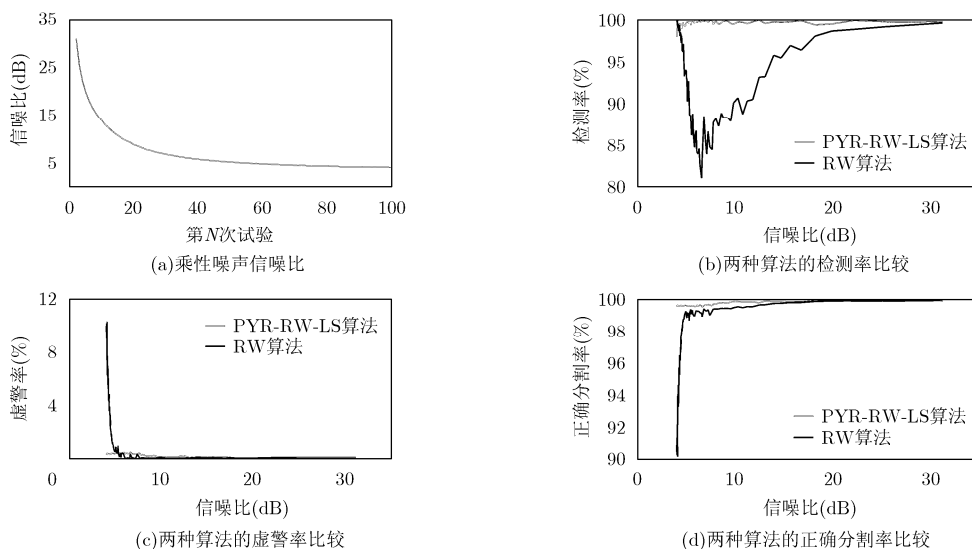


图 7 乘性噪声的抗噪性实验结果

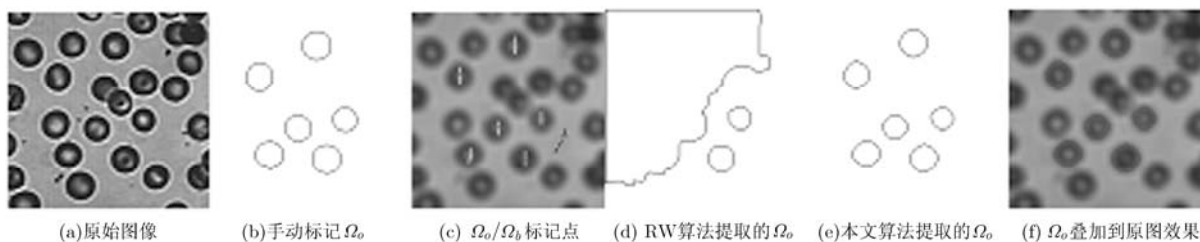


图 8 模糊噪声的实验结果

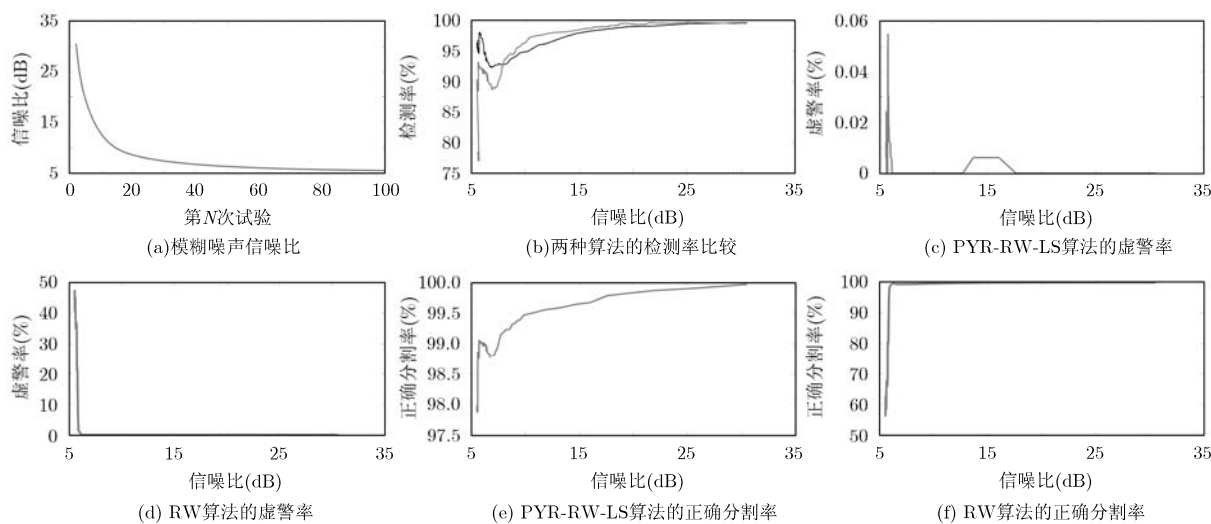


图 9 模糊噪声的抗噪性实验结果

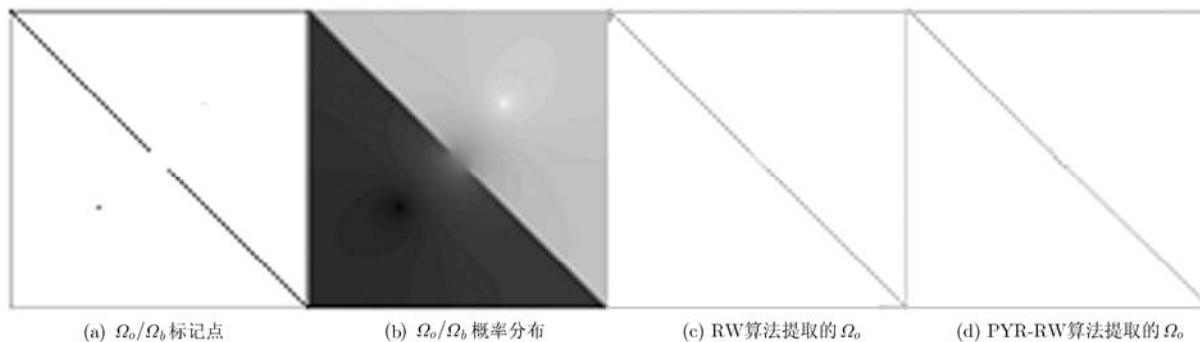


图10 弱边缘敏感性测试

表3 图像清晰度

测试图像(分辨率)	金字塔各层图像方差			
	第1层图像	第2层图像	第3层图像	第4层图像
Chest 图像(513×512)	151.7366(513×512)	75.5251(256×256)	23.5604(128×128)	8.9990(64×64)
Human 图像(320×240)	34.0086(320×240)	16.9171(160×120)	1.8578(80×60)	1.5133(40×30)
Blood 图像(272×265)	54.3086(272×265)	26.6936(136×132)	8.6915(68×66)	4.3509(34×33)

## 参考文献

- [1] Campbell N D F, Vogiatzis G, Hernández C, *et al.*. Automatic 3D object segmentation in multiple views using volumetric graph-cuts[J]. *Image and Vision Computing*, 2010, 28(1): 14-25.
- [2] Ding Lei and Yilmaz A. Interactive image segmentation using probabilistic hypergraphs[J]. *Pattern Recognition*, 2010, 43(5): 1863-1873.
- [3] Grady L. Random walks for image segmentation[J]. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2006, 28(11): 1768-1783.
- [4] Merris R. Laplacian matrices of graphs: a survey[J]. *Linear Algebra and Its Applications*, 1994, 197(1): 143-176.
- [5] Cheng Ming-ming and Zhang Guo-xin. Connectedness of random walk segmentation[J]. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2011, 33(1): 200-202.
- [6] Lai Yu-kun, Hu Shi-min, and Martin Ralph R. Rapid and effective segmentation of 3D models using random walks[J]. *Computer Aided Geometric Design*, 2009, 26(6): 665-679.
- [7] Kushnir D, Galun M, and Brandt A. Efficient multilevel eigensolvers with applications to data analysis tasks[J]. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2010, 32(8): 1377-1391.
- [8] Diestel R. Graph Theory[M]. 4th Ed. German: Springer-Verlag, 2010: 451-471.
- [9] Shi Yong-gang and Karl W C. A real-time algorithm for the approximation of level-set-based curve evolution[J]. *IEEE Transactions on Image Processing*, 2008, 17(5): 645-656.
- [10] 孔丁科, 汪国昭. 基于局部图划分多相活动轮廓图像分割模型[J]. 电子与信息学报, 2010, 32(9): 2127-2132.
- Kong Ding-ke and Wang Guo-zhao. Localized graph-cuts based multiphase active contour model for image segmentation[J]. *Journal of Electronics & Information Technology*, 2010, 32(9): 2127-2132.
- [11] Shih Y F. Image Processing and Pattern Recognition: Fundamentals and Techniques[M]. USA, John Wiley&Sons, Inc, Hoboken: New Jersey, 2010: 1-350.
- [12] Fawcett T. An introduction to ROC analysis[J]. *Pattern Recognition Letters*, 2006, 27(82): 861-874.
- 汪云飞: 男, 1985年生, 博士, 研究方向为图像分割与目标识别。  
 毕笃彦: 男, 1962年生, 教授, 研究方向为图像处理与模式识别。  
 史德琴: 女, 1970年生, 副教授, 研究方向为图像处理。  
 黄天军: 男, 1985年生, 助理工程师, 研究方向为图像处理。  
 刘迪: 男, 1988年生, 硕士, 研究方向为图像分割与目标识别。