

## 基于水平集方法的交互式目标分割算法

朴春赫, 赵海, 朱宏博, 徐久强

(东北大学信息科学与工程学院, 沈阳 110819)

**摘要:** 研究以用户与系统之间的交互信息为先验知识的目标分割算法, 提出一种基于改进速度函数的水平集图像分割算法。利用待分割图像的区域信息与目标轮廓信息, 改进传统 CV 模型中的速度函数。使用图像边缘的梯度下降向量逐步逼近目标边缘, 采用概率密度函数使图像中的所有像素值近似满足高斯概率分布, 并通过边界收敛模型保证速度函数快速收敛至目标边缘。实验结果表明, 对于包含噪声和目标边缘模糊的图像, 与基于 CV 模型的分割算法和基于均方差水平集方法的分割算法相比, 该算法具有更好的分割性能和鲁棒性。

**关键词:** 图像分割; 交互式分割; 水平集; 主动轮廓模型; 曲线演化

**中文引用格式:** 朴春赫, 赵海, 朱宏博, 等. 基于水平集方法的交互式目标分割算法[J]. 计算机工程, 2016, 42(2): 229-235.

**英文引用格式:** Pak Chun-hyok, Zhao Hai, Zhu Hongbo, et al. Interactive Object Segmentation Algorithm Based on Level Set Method[J]. Computer Engineering, 2016, 42(2): 229-235.

## Interactive Object Segmentation Algorithm Based on Level Set Method

Pak Chun-hyok, ZHAO Hai, ZHU Hongbo, XU Jiuqiang

(College of Information Science and Engineering, Northeastern University, Shenyang 110819, China)

**[Abstract]** This paper focuses on an object segmentation algorithm based on the interactive information as the prior knowledge between user and system, and then proposes a level set object segmentation algorithm based on improved speed function. It improves speed function of Chan & Vese (CV) model by regional and contour information. The input contour is gradually close to the actual one by using gradient descent vector. Probability density model ensures all the pixels in the images approximate Gaussian probability distribution. To speed-up the convergence, it utilizes the boundary of convergence model. Experimental results show that this algorithm has the better segmentation performance and robustness than the segmentation algorithms based on CV model and Mean Square Error (MSE) level set for the images including noise and object edge blur.

**[Key words]** image segmentation; interactive segmentation; level set; active contour model; curve evolution

**DOI:** 10.3969/j.issn.1000-3428.2016.02.041

### 1 概述

图像分割是图像处理领域的难点问题。目前, 自动分割方法虽然在某些领域取得了一定的研究成果, 但还远不能满足图像处理领域对分割结果准确性的要求。因此, 近年来由用户参与控制和引导的交互式分割方法在医学图像分割中受到越来越多的关注<sup>[1]</sup>。为了提高图像分割准确率, 图像分割算法应充分体现待分割图像的特征。现有的图像分割算法有阈值法、并行微分法、并行微分算子法、串行边界查找法、区域生长法、水平集方法、聚类方法等<sup>[2]</sup>。

文献[3]提出基于主动轮廓模型的水平集图像分割算法, 通过图像拓扑结构的迭代运算来演化分割曲线, 然后进行图像分割, 具有较好的稳定性。文献[4-5]提出基于图像边缘的水平集图像分割方法。但由于该方法只能使曲线以预设的形态演化, 曲线演化过程中经常需要扩张或收缩。上述方法都是基于图像边缘信息设计速度函数, 从而导致分割方法对噪声不够鲁棒, 同时在弱边缘处的效果并不理想。主动轮廓水平集方法中还有基于区域平均的图像分割算法<sup>[6]</sup>和基于区域均方差的图像分割算法<sup>[7]</sup>。基于区域平均的图像分割算法<sup>[6]</sup>采用分段函数并令其

**基金项目:** 国家科技支撑计划基金资助项目(2012BAH82F04)。

**作者简介:** 朴春赫(1976-), 男, 博士研究生, 主研方向为计算机视觉、图像处理; 赵海, 教授; 朱宏博, 博士研究生; 徐久强, 教授。

**收稿日期:** 2014-07-08 **修回日期:** 2015-01-22 **E-mail:** stu.hongbo.zhu@gmail.com

逼近待分割图像,通过最小化误差来实现图像分割<sup>[8-9]</sup>,由于其使用基于图像的区域信息设计的能量函数,因此对噪声具有较强的鲁棒性,同时对弱边缘目标也有很好的分割效果。但上述基于区域平均或均方差的水平集方法只是通过计算区域的平均值或寻找均方差演化分割曲线,所以,难以实现目标轮廓上的凹部分或相似像素值部分的分割。文献[10-11]在考虑目标轮廓信息分割的基础上,提出利用图像边缘梯度向量的梯度向量流(Gradient Vector Flow, GVF)模型。文献[12]通过加入动态标签函数限制了形状先验在图像中的作用范围。然而,上述方法仅适用于单先验的情况,而对于多先验的情况并不适用。为此,文献[13]利用边缘与区域信息,对先验水平集图像进行分割,但是该方法的分割图像上有很多噪声,且不适用于边缘模糊的图像分割。

为解决以上问题,本文提出一种基于水平集方法的交互式目标分割算法。该算法的基本思想为:设计一种新的速度函数替代 CV(Chan & Vese)模型中的速度函数,该函数考虑了给定图像的区域信息和目标轮廓信息,以及为了减少目标周围噪声影响的规整项,利用图像的边缘梯度向量推导出目标轮廓信息。通过对每个区域内像素值的统计分析进行图像分割,且该像素值的分布满足高斯分布。

## 2 水平集方法

水平集方法是一种用于界面追踪和形状建模的数值技术。水平集方法的优点为可以在笛卡尔网格上对演化中的曲线、曲面进行数值计算而不必对曲线、曲面进行参数化(即欧拉法),并且可以方便地追踪物体的拓扑结构变化。也就是说,在水平集模型中,用户选择的原始轮廓随着时间逐渐演化到目标轮廓。图1表示使用水平集方法的轮廓进化过程<sup>[3]</sup>,其中, $C(t)$ 是随着时间 $t$ 的动态轮廓; $\phi(x,y,t)$ 是 $(x,y)$ 随着时间 $t(t=0,1,2)$ 变化的水平值。

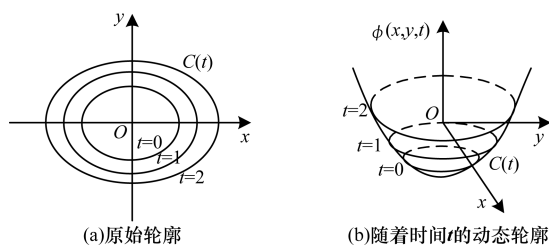


图1 使用水平集方法的轮廓进化过程

定义 $\Omega \subseteq R^2$ 为待分割的二位图像, $C$ 为动态轮廓。图像共分为 $C$ 、内部和外部3个部分,具体表示如下: $C = \partial\omega$ , $\omega = \text{inside}(C)$ , $\Omega/\omega = \text{outside}(C)$ 。对每个像素定义一个水平值(见图2,箭头表示动态轮廓扩展的方向),在轮廓 $C$ 内部水平值大于0,外部小于0,轮廓 $C$ 上等于0。其中,动态轮廓 $C$ 以给定的速度在法线( $F$ )方向上扩展直至输入图像的边缘。

廓扩展的方向),在轮廓 $C$ 内部水平值大于0,外部小于0,轮廓 $C$ 上等于0。其中,动态轮廓 $C$ 以给定的速度在法线( $F$ )方向上扩展直至输入图像的边缘。

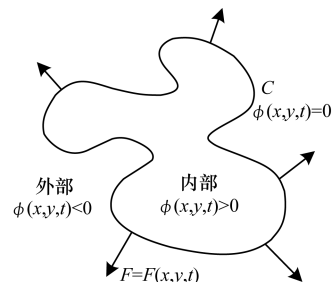


图2 轮廓 $C$ 沿法线方向扩展的示意图

在水平集方法中<sup>[6]</sup>,用 Lipschitz 函数 $\phi$ 的零水平表示轮廓 $C$ :

$$\begin{cases} C = \partial\omega = \{(x,y) \in \Omega: \phi(x,y) = 0\} \\ \text{inside}(C) = \omega = \{(x,y) \in \Omega: \phi(x,y) > 0\} \\ \text{outside}(C) = \Omega/\omega = \{(x,y) \in \Omega: \phi(x,y) < 0\} \end{cases} \quad (1)$$

随着时间 $t$ 的变化,函数 $\phi$ 的变化如下:

$$\begin{cases} \phi(x,y,t) = 0 \\ \phi_0(x,y) = \phi(x,y,0) \end{cases} \quad (2)$$

其中, $\phi_0(x,y)$ 为用户输入的最初轮廓。在式(2)中,零水平函数 $\phi$ 随着时间 $t$ 的变化,通过链式法则表示如下:

$$\frac{\partial\phi(x,y,t)}{\partial t} + \nabla\phi(x,y,t) \frac{\partial\omega}{\partial t} = 0 \quad (3)$$

其中,定义法线方向的速度函数 $F(t)$ 为:

$$F(t) = \frac{\partial\omega}{\partial t} \mathbf{n} \quad (4)$$

其中, $\mathbf{n}$ 是轮廓的法线方向向量;水平函数 $\phi$ 的梯度表示如下:

$$\mathbf{n} = \frac{\nabla\phi(x,y,t)}{|\nabla\phi(x,y,t)|} \quad (5)$$

由式(4)和式(5)可得:

$$F(t) |\nabla\phi(x,y,t)| = \nabla\phi(x,y,t) \frac{\partial\omega}{\partial t} \quad (6)$$

由式(3)和式(6)可得,水平函数 $\phi$ 可由如下的偏微分方程(Partial Differential Equation, PDE)表示:

$$\frac{\partial\phi(x,y,t)}{\partial t} + F(t) |\nabla\phi(x,y,t)| = 0 \quad (7)$$

所以,在任意时刻法线方向上的速度函数决定了轮廓上任意点 $(x,y)$ 沿法线方向上的变化量。使用动态轮廓的微分形式,法线方向速度函数 $F(t)$ 表示如下:

$$\frac{\partial C}{\partial t} = F \cdot \mathbf{n} \quad (8)$$

由上面分析可知,速度函数 $F(t)$ 会对目标分割的结果产生影响。

### 3 用于分割二位灰度图像的水平集方法

#### 3.1 CV 模型

CV 模型是一种较为经典的图像分割模型<sup>[6]</sup>。其主要思想为:利用动态轮廓分割区域的平均值产生能量函数,通过最小化能量函数实现目标分割。CV 模型的能量函数表示如下:

$$\begin{cases} E^{CV}(c, c_1, c_2) = \mu \cdot \text{Length}(C) \\ \quad + \nu \cdot \text{Area}(\text{inside}(C)) \\ \quad + \lambda_1 \int_{\text{outside}(C)} |I(x, y) - c_1| dx dy \\ \quad + \lambda_2 \int_{\text{inside}(C)} |I(x, y) - c_2| dx dy \\ c_1 = \text{average}(I(x, y), \text{outside}(C)) \\ c_2 = \text{average}(I(x, y), \text{inside}(C)) \end{cases} \quad (9)$$

其中,  $c$  是动态轮廓;  $c_1$  是动态轮廓外部像素值的平均值;  $c_2$  是动态轮廓内部像素值的平均值;  $I(x, y)$  是  $(x, y)$  的像素值;  $\mu, \nu, \lambda_1, \lambda_2 \geq 0$ , 在实际计算中,  $\lambda_1 = \lambda_2 = 1, \nu = 0$ 。式(9)水平集的微分表示如下:

$$\frac{\partial \phi}{\partial t} = \delta_\epsilon(\phi) \begin{bmatrix} \mu \cdot \text{div} \left( \frac{\nabla \phi}{|\nabla \phi|} \right) - \nu \\ -\lambda_1 (I - c_1)^2 + \lambda_2 (I - c_2)^2 \end{bmatrix} \quad (10)$$

其中,  $\delta_\epsilon$  表示 Heaviside 函数的导数。

如果  $\lambda_1 = \lambda_2 = 1, \mu = \nu = 0$ , CV 模型中每个点的法线方向上的速度函数表示如下:

$$F = 2\delta_\epsilon(\phi) (c_1 - c_2) \left( I(x, y) - \frac{c_1 + c_2}{2} \right) \quad (11)$$

#### 3.2 基于均方差的水平集方法

文献[7]提出一种基于均方差 (Mean Square Error, MSE) 的水平集方法。该方法假设动态轮廓的内部及外部像素值分布为高斯分布,因此,用均方差可以把动态轮廓逼近到目标边缘。基于均方差的水平集函数表示如下:

$$\begin{cases} \frac{\partial \phi}{\partial t} = \delta_\epsilon(\phi) \left( \log \frac{\sigma_1}{\sigma_2} - \frac{(I - C_2)^2}{\sigma_2} + \frac{(I - C_1)^2}{\sigma_1} \right) \\ C_1 = \text{average}(I), \sigma_1 = \text{deviation}(I), \text{outside}(C) \\ C_2 = \text{average}(I), \sigma_2 = \text{deviation}(I), \text{inside}(C) \end{cases} \quad (12)$$

其中,  $\sigma_1$  表示动态轮廓外部像素值的均方差;  $\sigma_2$  表示动态轮廓内部像素值的均方差;  $C_1$  是动态轮廓外部像素值的平均值;  $C_2$  是动态轮廓内部像素值的平均值。

由式(12)可得,速度函数表示如下:

$$F = \log \frac{\sigma_1}{\sigma_2} - \frac{(I - C_2)^2}{\sigma_2} + \frac{(I - C_1)^2}{\sigma_1} \quad (13)$$

基于 CV 模型的分割算法是经典的基于动态轮廓的图像分割算法,但由于其 CV 模型容易出现局部最优化,且时间复杂度较高,因此不适用于对时效性要

求较高的应用场景。此外,基于均方差的水平集方法<sup>[7]</sup>能够保证在运动方向上的动态轮廓清晰准确。

### 4 基于水平集方法的交互式目标图像分割

#### 4.1 算法描述

交互式目标分割过程需要利用用户输入的最初轮廓。因此,该算法应该包含初期输入轮廓特征和图像中的目标特征。

对于输入初始轮廓的情况,只有当输入的初始轮廓比较接近目标轮廓时算法才有效。但由于目标轮廓的复杂性,对于初始轮廓的输入来说比较困难,比如,目标轮廓是分开的或者在内部有环。因此需设计初始轮廓不依赖目标轮廓位置或者形态的分割模型。为了使动态轮廓尽快逼近目标轮廓,本文模型需要包含目标轮廓信息,利用基于像素值的梯度向量使附近的动态轮廓尽快地移动至目标轮廓。但如果仅使用基于像素值的梯度向量,则只有在像素值变化很大时该向量才有效。也就是说,动态轮廓在靠近目标轮廓时收敛性很低。为了克服该缺点,本文引入梯度向量流 (GVF) 模型<sup>[10-11]</sup>。该模型由图像边缘图的梯度向量组成。GVF 向量场为:  $V_{GVF}(x, y) = \{(u(x, y), v(x, y))\}$ , 其中,  $u(x, y)$  和  $v(x, y)$  分别是边缘图  $x$  和  $y$  方向的梯度值分量。通过最小化式(14)的能量函数求出<sup>[14]</sup>。

$$E = \iint \{ \mu (u_x^2 + v_x^2 + u_y^2 + v_y^2) + [|\nabla f|^2] V - \nabla f \} dx dy \quad (14)$$

其中,  $u_x, u_y, v_x$  和  $v_y$  为场的空间导数;  $f(x, y)$  为边缘图;  $\nabla f$  为边缘图的梯度;  $\mu (0 < \mu < 1)$  为调整因子,噪声越多,其值越大。满足上述条件的  $u(x, y)$  和  $v(x, y)$  可用 Euler 方程迭代求出。

$$\begin{cases} \mu \nabla^2 u_x - (u_x - f_x)(f_x^2 + f_y^2) = 0 \\ \mu \nabla^2 v_x - (v_x - f_x)(f_x^2 + f_y^2) = 0 \\ u_x^{(0)} = \frac{\partial f}{\partial x} \\ v_x^{(0)} = \frac{\partial f}{\partial y} \end{cases} \quad (15)$$

其中,  $f_x$  和  $f_y$  为边缘图  $f(x, y)$  的  $x$  方向和  $y$  方向的分量。

图像上每个点的 GVF 方向都指向目标轮廓,当它靠近目标轮廓时值很大,靠近像素值变化不大的区域时值很小,此时能量函数表示如下<sup>[14]</sup>:

$$E_1 = \int_C (V_{GVF} \otimes \mathbf{n}) dx dy \quad (16)$$

其中,  $\mathbf{n}$  是法线向量。利用水平集方法求得的法线方向速度函数具体如下:

$$F_1 = \text{sign}(V_{GVF} \otimes \nabla \phi) \text{div}(V_{GVF}) \quad (17)$$

其中,  $div(V_{GVF})$  表示在  $V_{GVF}$  区域内的一个点上场的散度。

当动态轮廓移动到附近目标时, 观察动态轮廓的移动情况。在图像中给定目标状态具有多样性, 比如, 假设目标的轮廓是模糊的, 那么在不考虑图像区域信息的情况下, 得到的结果会产生收缩现象, 甚至被收缩到一个点。所以, 本文利用图像区域信息, 得到正确的目标轮廓。为了得到图像区域信息, 做如下假设: 在动态轮廓外部及内部, 图像灰度值的分布为相互独立的高斯分布。这样动态轮廓内外部之间的高斯概率密度函数表示如下:

$$p(I | \{\Omega_1, \Omega_2\}) = p(I | \Omega_1) \cdot p(I | \Omega_2) \\ = \prod_{\omega \in \Omega_1} p_1(I(\omega_1)) \cdot \prod_{\omega \in \Omega_2} p_2(I(\omega_2)) \quad (18)$$

其中,  $\Omega_1$  和  $\Omega_2$  表示动态轮廓的外部 and 内部;  $p_1(I(\omega_1))$  和  $p_2(I(\omega_2))$  表示动态轮廓外部及内部的高斯概率密度函数。

对于给定图像, 当能量函数最小时, 图像分割结果最优。

$$E_2(\Omega) = \int_{\Omega_1} \log p_1(I(x, y)) dx dy \\ - \int_{\Omega_2} \log p_2(I(x, y)) dx dy \quad (19)$$

为了使式(19)的积分区域一致, 导入 Heaviside 函数, Heaviside 函数计算公式见式(20)。其几何学意义如图 3 所示。

$$H(x) = \begin{cases} 1 & x > 0 \\ 0 & x < 0 \end{cases} \quad (20)$$

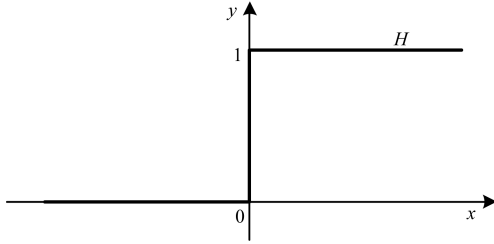


图 3 Heaviside 函数示意图

结合 Heaviside 函数, 式(19)可表示为:

$$E_2(\Omega) = \int_{\Omega} (1 - H_{(\phi)}) \log p_1(I(x, y)) dx dy \\ - \int_{\Omega} H_{(\phi)} \log p_2(I(x, y)) dx dy \quad (21)$$

其中,  $\phi$  是图像的水平集函数, 在动态轮廓内部  $\phi > 0$ , 在动态轮廓外部  $\phi < 0$ 。Euler 方程表示如下:

$$\frac{d\phi}{dt} = H'_{(\phi)} (\log p_2(I(x, y)) - \log p_1(I(x, y))) \quad (22)$$

包含图像区域信息的速度函数表示如下:

$$F_2 = \log p_2(I(x, y)) - \log p_1(I(x, y)) \quad (23)$$

高斯概率密度函数  $p(I(x, y))$  表示如下:

$$p_1(\Omega_1) = \frac{1}{\sqrt{2\pi}\sigma_1} \cdot \exp\left(-\frac{(I-C_1)^2}{2\sigma_1^2}\right) \quad (24)$$

$$p_2(\Omega_2) = \frac{1}{\sqrt{2\pi}\sigma_2} \cdot \exp\left(-\frac{(I-C_2)^2}{2\sigma_2^2}\right) \quad (25)$$

其中,  $C_1$  和  $C_2$ ,  $\sigma_1$  和  $\sigma_2$  表示动态轮廓外部和内部的高斯概率密度函数的标准方差和平均值。

根据  $p(I(x, y))$  的展开式, 式(23)可表示为:

$$F_2 = \log\left(\frac{\sigma_1}{\sigma_2}\right) - \frac{(I-C_2)^2}{2\sigma_2} + \frac{(I-C_1)^2}{2\sigma_1} \quad (26)$$

在添加规整项后的速度函数  $F_2$  表示如下:

$$F_2 = \log\left(\frac{\sigma_1}{\sigma_2}\right) - \frac{(I-C_2)^2}{2\sigma_2} \\ + \frac{(I-C_1)^2}{2\sigma_1} + \nu \cdot \operatorname{div}\left(\frac{\nabla\phi}{|\nabla\phi|}\right) \quad (27)$$

结合轮廓信息和区域信息, 总体速度函数如下:

$$F = \alpha F_1 + \beta F_2 = \alpha (\operatorname{sign}(V_{GVF} \otimes \nabla\phi) \operatorname{div}(V_{GVF})) \\ + \beta \left( \log\left(\frac{\sigma_1}{\sigma_2}\right) - \frac{(I-C_2)^2}{2\sigma_2} \right. \\ \left. + \frac{(I-C_1)^2}{2\sigma_1} + \nu \cdot \operatorname{div}\left(\frac{\nabla\phi}{|\nabla\phi|}\right) \right) \quad (28)$$

其中,  $\alpha$  和  $\beta$  随着图像的状态不同而取不同的值, 图像内噪声越多值越大,  $\alpha$  和  $\beta$  之间的关系为  $\alpha + \beta = 1$ 。

#### 4.2 算法实现

交互式目标分割过程就是基于用户输入的初期轮廓到目标轮廓的迭代过程。轮廓进化的正确性与其速度函数有关。本文提出的速度函数如式(28)所示。在具体实现过程中, 使用近似函数  $H_\varepsilon$  代替 Heaviside 函数  $H_{(\phi)}$ , 近似函数  $H_\varepsilon$  的具体形式如下<sup>[6]</sup>:

$$H_\varepsilon(\phi) = \frac{1}{2} \left[ 1 + \frac{2}{\pi} \arctan\left(\frac{\phi}{\varepsilon}\right) \right] \quad (29)$$

其中,  $\varepsilon = 1$ 。  $\delta_\varepsilon(\phi)$  具体的形式如下<sup>[6]</sup>:

$$\delta_\varepsilon(\phi) = H'_\varepsilon(\phi) = \frac{1}{\pi} \frac{\varepsilon}{\varepsilon^2 + \phi^2} \quad (30)$$

其中, 轮廓内外部的平均值  $C_i$  ( $i = 1, 2$ ) 和标准差可用 Heaviside 函数分别表示如下:

$$\begin{cases} C_1 = \frac{\int I \cdot (1 - H) dx dy}{\int (1 - H) dx dy} \\ C_2 = \frac{\int I \cdot H dx dy}{\int H dx dy} \end{cases} \quad (31)$$

$$\begin{cases} \sigma_1^2 = \frac{\int (I - C_1)^2 \cdot (1 - H) dx dy}{\int (1 - H) dx dy} \\ \sigma_2^2 = \frac{\int (I - C_2)^2 \cdot H dx dy}{\int H dx dy} \end{cases} \quad (32)$$

现有的水平集更新方法如下:

$$\phi^{t+1} = \phi^t + \Delta t \cdot F \quad (33)$$

其中,  $\phi^t$  是当前的水平集;  $\phi^{t+1}$  是更新水平集;  $\Delta t$  是时间增量。

本文算法步骤具体如下:

**Step1** 输入初期轮廓,在初期轮廓的基础上,决定初期水平集。

**Step2** 计算向量场  $V_{GVF}$ 。

**Step3** 使用 Heaviside 近似函数式(29)计算图像上每一个点的 Heaviside 近似值。

**Step4** 使用当前水平集和每一个点的 Heaviside 近似值,更新前一轮廓(初始时,为用户输入的轮廓)内外的图像像素值平均值( $C$ )。

**Step5** 使用计算的轮廓内外的图像像素值平均值和 Heaviside 近似函数值,更新前一轮廓(初始时,为用户输入的轮廓)内外的图像像素值标准差( $\sigma$ )。

**Step6** 使用式(26)计算考虑区域信息的速度分量。

**Step7** 使用已计算的向量场  $V_{GVF}$  和考虑区域信息的速度分量,计算总体的速度函数。

**Step8** 更新当前的水平集。

**Step9** 循环 Step3 ~ Step8,直到满足给定条件。

本文算法流程如图4所示。

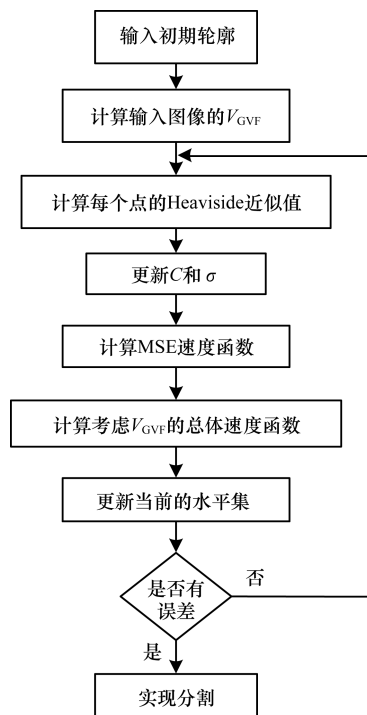


图4 本文算法流程

## 5 实验结果与分析

实验分为2组:(1)在性能方面对本文提出算法和已有的水平集算法进行比较;(2)使用本文算法对

实际图像进行分割并且分析图像分割准确性。

对于人工构造的图像,分别应用基于 CV 模型的分割算法(简称为 CV)、基于均方差(MSE)水平集的分割算法(简称为 MSE 算法)和本文算法进行图像分割。人工构造的图像有2种:一种是边界模糊的图像;另一种是添加了噪声的图像。进入实验阶段之前,对于边界清晰和无噪声的图像,利用3种算法进行图像分割效果对比,如图5所示。可以看出,3种算法对于人工构造图像,可以获得较为理想的图像分割效果。图6为图5的细节放大图,在局部细节信息尤其是边界细节上,若图像的边缘清晰,CV与MSE均可以获得较为理想的分割效果。CV分割结果稍优于MSE算法与本文算法,主要是由于CV在计算过程中使用分割区域的像素灰度平均值。

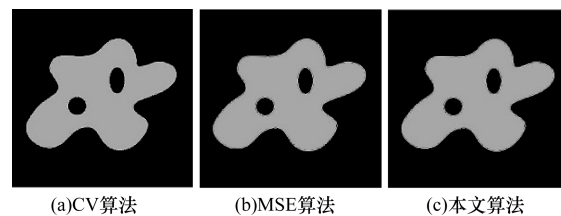


图5 对于理想图像的分割效果比较

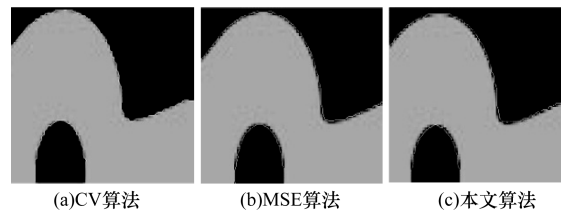


图6 对于理想图像细节放大后的部分分割效果比较

图7表示对边界模糊的图像应用3种算法的图像分割结果。可以看出,CV速度较快,但分割算法的分割效果并不理想。

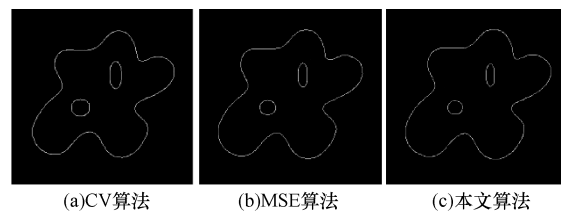


图7 对于边界模糊图像的分割效果比较

图8是放大图7目标边界的一部分图像,分割轮廓略显收缩。一般来说,图像处理之前需要预处理,使用低通滤波器消除图像噪声。因此,导致目标图像的边界会略显模糊。在使用 CV 模型分割目标时,其分割结果会略显收缩。但是不论使用基于 MSE 的水平集方法还是使用本文算法,目标分割结

果都是正确的。其因为 CV 模型是基于分割区域图像的平均像素值进行分割,而 MSE 和本文算法使用图像区域信息(像素值的高斯分布)。所以,利用 MSE 和本文算法的图像分割准确性高于 CV 模型。表 1 为 3 种算法的图像轮廓收缩率。收缩率是指未模糊图像中的目标面积和图像分割后目标面积之比。未模糊图像中的目标面积利用 CV 模型的结果进行计算。

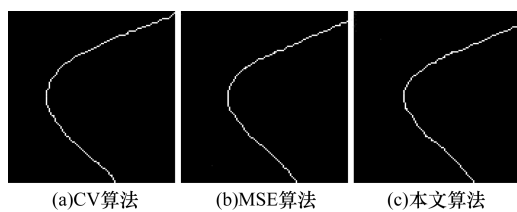


图 8 对于边界模糊图像细节放大后的部分分割效果比较

表 1 轮廓收缩率比较 %

图像分割算法	收缩率
CV 算法	4.2
MSE 算法	0.5
本文算法	0.4

从表 1 可以看出, MSE 和本文算法的图像轮廓收缩率较小。图 9 表示含有噪声的图像分割结果。可以看出,使用本文算法进行图像分割的准确性较好且速度较快。另外,噪声在前景区域和背景区域分布比较均匀。当使用 CV 时,最终分割结果中的前景区域上的噪声量多于背景区域。当使用 MSE 时,消除了总体噪声的影响,但其在目标区域的噪声消除能力差于本文算法。特别地,当使用本文算法时,约 83 s 后目标分割误差达到给定的误差范围,但使用 MSE 时,约 117 s 后能达到给定的误差范围。CV 在迭代次数超过 20 次、计算时间超过 343 s 后,仍没有收敛至给定的误差范围内。所以,在综合考虑目标边界的图像分割特性和噪声消除能力上,本文算法性能较优。

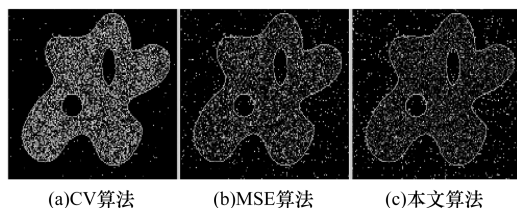


图 9 添加噪声后的图像分割效果比较

图 10 表示不同程度噪声情况下的图像分割结果。第 1 列~第 3 列分别为 CV、MSE 和本文算法。从图 9 和 10 可以看出,图像越复杂, CV 分割

效果越差。MSE 算法和本文算法的分割效果比较接近。

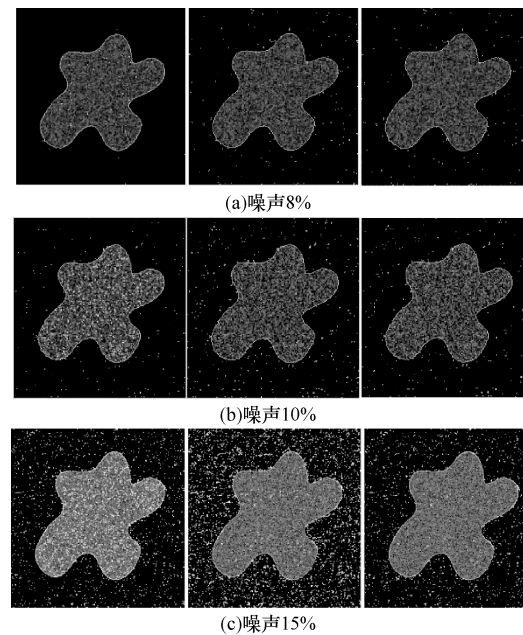


图 10 对于噪声图像的分割效果比较

表 2 表示 3 种算法的噪声消除能力比较。通过计算相同分割时间内噪声像素量的方法评价噪声消除能力。图 11 表示使用高斯滤波器情况下的图像分割结果。此时,  $\alpha = 0.6$ , 预处理高斯内核的尺度为 2.0。从图 11 可以看出,使用同样的高斯滤波器时,本文算法可以正确地分割目标图像,而 CV 和 MSE 的分割结果中仍然存在噪声。

表 2 噪声消除率比较 %

噪声量	CV 算法	MSE 算法	本文算法
8	96.34	84.78	86.96
10	77.48	79.36	83.23
15	67.80	69.51	77.58

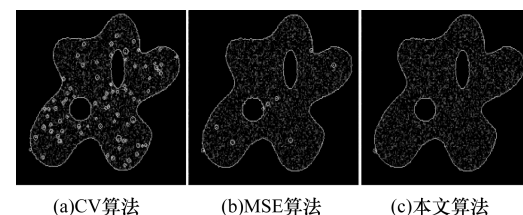


图 11 基于高斯滤波器的图像分割效果比较

CV 和本文算法对于医学图像的分割结果如图 12 所示。可以看出,本文算法比 CV 更有效,因为使用 CV 进行图像分割的结果收缩较大,而本文算法对模糊图像和噪声图像均具有较好的分割能力。使用含有多个待分割目标的图像评价本文分割算法的鲁棒性,如图 13 所示,其中,图像分割结果图中灰色区域是

初始轮廓;白色线段包围的区域是得到的目标轮廓。最左边图像为原图像,其他图像为不同用户输入轮廓的图像分割结果。实验结果表明,本文算法不会受到用户输入轮廓和原图像形状的影响。

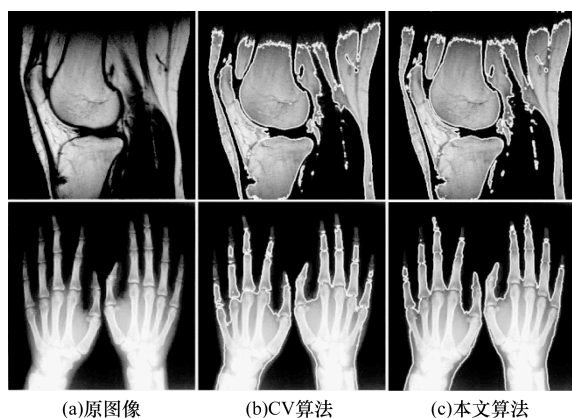


图12 对于医学图像的图像分割效果比较

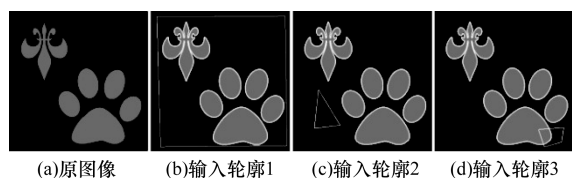


图13 本文算法对包含多个目标图像的分割效果

## 6 结束语

本文提出一种基于改进速度函数的水平集图像分割算法,该速度函数利用待分割图像的区域信息和目标轮廓信息。使用图像边缘的梯度下降向量逐步逼近目标边缘,相较于其他算法具有更快的分割速度。采用概率密度模型使图像中的所有像素值近似满足高斯概率分布,通过边界收敛模型保证速度函数能够快速收敛至目标边缘。仿真实验结果表明,对于边界清晰的无噪声图像,CV具有较快的速度及较高的准确性。但在图像比较复杂的情况下,CV与MSE均受到不同程度的噪声干扰导致图像模糊边缘的分割结果不理想。相比之下,本文算法由于引入了图像的区域信息和轮廓信息从而对噪声图像和模糊图像的处理效果更为理想。在今后工作中将重点对算法的时间复杂度进行研究。另外,本文算法所提取的目标轮廓未考虑多闭合边界间的结构关系,无法直接结合医学知识对病例进行判定,这部分内容也会在今后工作中加以补充。

## 参考文献

- [1] Zhang Tao, Freedman D. Tracking Objects Using Density Matching and Shape Priors[C]//Proceedings of the 9th IEEE International Conference on Computer Vision. Washington D. C., USA: IEEE Press, 2003: 1056-1062.
- [2] 王雪,李宣平,戴逸翔.多相水平集协同空间模糊聚类图像多目标分割[J].机械工程学报,2013,49(20): 10-15.
- [3] Osher S,Paragios N. Geometric Level Set Methods in Imaging, Vision, and Graphics [M]. Berlin, Germany: Springer-Verlag,2003.
- [4] Malladi R, Sethian J A, Vemuri B C. Shape Modeling with Front Propagation:A Level Set Approach[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence,1995,17(2):158-175.
- [5] Caselles V, Kimmel R, Sapiro G. Geodesic Active Contours[J]. International Journal of Computer Vision, 1997,22(1):61-79.
- [6] Chan T F, Vese L A. Active Contours Without Edges[J]. IEEE Transactions on Image Processing, 2001,10(2):266-277.
- [7] Rousson M, Deriche R. A Variational Framework for Active and Adaptive Segmentation of Vector Valued Images[C]//Proceedings of 2002 Workshop on Motion and Video Computing. Washington D. C., USA: IEEE Press,2002:56-61.
- [8] Mumford D, Shah J. Optimal Approximations by Piecewise Smooth Functions and Associated Variational Problems [J]. Communications on Pure and Applied Mathematics,1989,42(5):577-685.
- [9] Mumford D,Shah J. Boundary Detection by Minimizing Functionals[J]. Image Understanding, 1988, 53 (2): 19-43.
- [10] Xu Chenyang,Prince J L. Snakes, Shapes, and Gradient Vector Flow [J]. IEEE Transactions on Image Processing,1998,7(3):359-369.
- [11] Chuang Cheng-hung, Lie Wen-nung. A Downstream Algorithm Based on Extended Gradient Vector Flow Field for Object Segmentation[J]. IEEE Transactions on Image Processing,2004,13(10):1379-1392.
- [12] Cremers D, Sochen N, Schnorr C. A Multiphase Dynamic Labeling Model for Variational Recognition-driven Image Segmentation[J]. International Journal of Computer Vision,2006,66(1):67-81.
- [13] 王斌,李洁,高新波.一种基于边缘与区域信息的先验水平集图像分割方法[J].计算机学报,2012, 35(5):1067-1071.
- [14] Jayadevappa D, Kumar S S, Murty D S. A New Deformable Model Based on Level Sets for Medical Image Segmentation[J]. IAENG International Journal of Computer Science,2009,36(3):1-9.

编辑 陆燕菲