



哈爾濱工業大學
HARBIN INSTITUTE OF TECHNOLOGY

图像处理

Image Processing

任课教师： 杨云云



02 GCRSF模型

B a c k g r o u n d a n d P u r p o s e



2.1 引言



由第1.2.4 节中对RSF模型的介绍可知，RSF模型引入一个核函数 K_0 ，该核函数 K_0 的局部性质使得RSF模型可以充分利用图像的局部强度信息。不同于CV模型用两个常量 c_1 和 c_2 来近似区域 $\text{inside}(C)$ 和 $\text{outside}(C)$ 内的图像强度，RSF模型用两个拟合函数 $f_1(x)$ 和 $f_2(x)$ 局部地近似轮廓线 C 两侧的图像强度。

因此RSF模型克服了CV模型的缺点，可以分割具有图像强度不均匀性质的图像。但是与CV模型类似，RSF模型也是非凸的，RSF模型的非凸性一方面会影响该模型的收敛速度，另一方面，非凸性也会导致该模型产生局部极小解，并且对初始轮廓线的选择很敏感。



2.1 引言



为了解决这些模型的非凸性带来的问题，Chan等人提出了GCS方法。GCS方法不仅使得模型在数值上更容易处理，而且使得模型不会陷入局部极小，从而更加可靠。Goldstein等人通过将GCS方法和Split Bregman方法应用于CV模型，提出一个凸的快速的分割模型，称为全局凸CV模型。但是与CV模型一样，全局凸CV模型主要用于分割具有同质区域的图像，而不可以分割具有图像强度不均匀性质的图像。

因此，为了更好地更快速地处理强度不均匀的图像，本章结合RSF模型，GCS方法和Split Bregman方法，提出一个新的快速的模型，称为全局凸区域可伸缩拟合能量模型，即**Globally Convex Region-Scalable Fitting Energy Model**，简称为**GCRSF模型**。**GCRSF模型**可以快速有效地处理具有图像强度不均匀性质图像的分割问题。



2.2 全局凸区域可伸缩拟合能量模型



基于RSF 模型能量泛函关于水平集函数 ϕ 的梯度下降流方程，为了应用全局凸分割方法和 Split Bregman 方法，需要下面几个步骤。

- 去掉 RSF 模型能量泛函梯度下降流方程的水平集正则项。
- 应用 GCS 方法的思想得到一个简化的梯度下降流方程，该简化的梯度流方程与去掉水平集正则项之后的梯度下降流方程有相同的静态解。
- 基于该简化的梯度流方程，定义一个新的能量泛函。



Li 等人已经由标准的梯度下降法给出 RSF 模型能量泛函 $\mathcal{F}_\varepsilon^{RSF}(\phi, f_1, f_2)$ 关于水平集函数 ϕ 的梯度下降流方程如下:

$$\frac{\partial \phi}{\partial t} = -\delta_\varepsilon(\phi)(\lambda_1 e_1 - \lambda_2 e_2) + \nu \delta_\varepsilon(\phi) \operatorname{div} \left(\frac{\nabla \phi}{|\nabla \phi|} \right) + \mu \left[\nabla^2 \phi - \operatorname{div} \left(\frac{\nabla \phi}{|\nabla \phi|} \right) \right] \quad (2-1)$$

其中 $e_i (i = 1, 2)$ 定义如下:

$$e_i(\mathbf{x}) = \int_{\Omega} K_\sigma(\mathbf{y} - \mathbf{x}) |u_0(\mathbf{x}) - f_i(\mathbf{y})|^2 d\mathbf{y} \quad (2-2)$$

其中 $f_1(\mathbf{x})$ 和 $f_2(\mathbf{x})$ 是用来近似轮廓线 C 两侧的区域 $\text{inside}(C)$ 和 $\text{outside}(C)$ 内图像强度的两个拟合函数。



2.2 全局凸区域可伸缩拟合能量模型



介绍上述梯度下降流方程(2-1)中右边三项的作用。

- $-\delta_\varepsilon(\phi)(\lambda_1 e_1 - \lambda_2 e_2)$ 对应于 RSF 能量泛函(1-25)中的数据拟合能量 $\mathcal{E}^{RSF}(\phi, f_1, f_2)$ 梯度下降流, 因此被称为数据拟合项。
- $\delta_\varepsilon(\phi) \operatorname{div} \left(\frac{\nabla \phi}{|\nabla \phi|} \right)$ 对应于式(1-25)中的长度能量 $L(\phi)$, 被称为长度项。其中数据拟合项和长度项这两项在水平集函数 ϕ 的演变过程中起到关键作用。数据拟合项驱使活动轮廓线朝向物体的边缘处运动, 长度项则对轮廓线具有长度缩短或光滑的作用, 从而保证轮廓线的正则性。
- $\left[\nabla^2 \phi - \operatorname{div} \left(\frac{\nabla \phi}{|\nabla \phi|} \right) \right]$ 由式(1-25)中的 $P(\phi)$ 导出, 被称为水平集正则项。 $P(\phi)$ 是由 Li 等人首次提出然后被用于 LBF 模型与 RSF 模型中。水平集正则项或者 $P(\phi)$ 的作用仅仅是消除重新初始化水平集函数 ϕ 的过程, 并保证水平集函数 ϕ 在零水平集附近是一个近似的符号距离函数。事实上, 水平集正则项对于迫使活动轮廓线朝向物体的边缘处演变起不到任何作用。



- 从方程(2-1)中去掉水平集正则项。

事实上，去掉该水平集正则项也保证了后面可以应用 Split Bregman 方法快速地极小化新提出的能量泛函。从方程(2-1)中去掉水平集正则项是合理的，并且不会影响新模型在实际中的可用性与可靠性。因此从方程(2-1)中去掉水平集正则项，并且不失一般性，取 $\nu = 1$ ，可得如下梯度流方程：

$$\frac{\partial \phi}{\partial t} = \delta_{\varepsilon}(\phi) \left[(-\lambda_1 e_1 + \lambda_2 e_2) + \operatorname{div} \left(\frac{\nabla \phi}{|\nabla \phi|} \right) \right] \quad (2-3)$$



2.2 全局凸区域可伸缩拟合能量模型



- 应用 GCS 方法的思想简化的梯度下降流方程。

由式(1-17)中 H_ε 的定义可知, H_ε 是 Heaviside 函数 H 的一个具有非紧支撑的光滑近似, 并且满足: 对于所有的 $z \in R$, 都有 $H_\varepsilon(z) \in (0, 1)$ 。 δ_ε 是 H_ε 的导数, 即光滑的 Dirac delta 函数, 由式(1-18)中 δ_ε 的定义可知, δ_ε 满足: 对于所有的 $z \in R$, 都有 $\delta_\varepsilon(z) \neq 0$ 。因此可以得到 $\delta_\varepsilon(\phi) \neq 0$ 处处成立。从而关于水平集函数 ϕ 的演变并不仅仅局限于它的零水平集 $\{\mathbf{x} : \phi(\mathbf{x}) = 0\}$ 附近的几条水平曲线, 而是被推广到水平集函数 ϕ 的所有水平集。根据全局凸分割方法的思想, 上述梯度下降流方程 (2-3) 与如下简化的梯度下降流方程有相同的静态解:

$$\frac{\partial \phi}{\partial t} = (-\lambda_1 e_1 + \lambda_2 e_2) + \operatorname{div} \left(\frac{\nabla \phi}{|\nabla \phi|} \right) \quad (2-4)$$

- 基于简化的梯度下降流方程 (2-4), 首先定义一个新的能量泛函。

$$E^{GCRSF}(\phi) = \int_{\Omega} |\nabla(\phi(\mathbf{x}))| d\mathbf{x} + \int_{\Omega} \phi(\mathbf{x})(\lambda_1 e_1(\mathbf{x}) - \lambda_2 e_2(\mathbf{x})) d\mathbf{x} \quad (2-5)$$



定义 2.1 标准的全变分 (Total Variation) 范数, 简称 TV 范数定义如下:

$$TV(\phi) = \int_{\Omega} |\nabla \phi(\mathbf{x})| d\mathbf{x} = |\nabla \phi|_1 \quad (2-6)$$

可以看出, 在式(2-5)中定义的能量泛函 $E^{GCRSF}(\phi)$ 的第一项正是标准的全变分范数, 为了更加容易地检测图像的边界, 将GAC 模型的思想用于新模型中, 即在提出的能量泛函(2-5)中引入一个非负的边缘检测函数 g 。这是通过用如下加权的全变分范数来代替式(2-6)中标准的全变分范数 $|\nabla \phi|_1$ 实现的。



定义 2.2 定义2.1中的标准的全变分范数的加权形式，即是加权的全变分范数定义如下：

$$TV_g(\phi) = \int_{\Omega} g(|\nabla u_0(\mathbf{x})|) |\nabla \phi(\mathbf{x})| d\mathbf{x} = |\nabla \phi|_g \quad (2-7)$$

其中边缘检测函数 g 在 GAC 模型中提到，定义如下：

$$g(\xi) = \frac{1}{1 + \beta |\xi|^2} \quad (2-8)$$

$\beta \geq 0$ 是一个用来决定分割的细节水平的常量。

于是，可以给出在式(2-5)中定义的能量泛函 $E^{GCRSF}(\phi)$ 的新的形式。



2.2 全局凸区域可伸缩拟合能量模型



定义 2.3 新提出的 GCRSF 模型的能量泛函 $E^{GCRSF}(\phi)$ 定义如下:

$$\begin{aligned} E^{GCRSF}(\phi) = & \int_{\Omega} g(|\nabla u_0(\mathbf{x})|)|\nabla(\phi(\mathbf{x}))|d\mathbf{x} \\ & + \int_{\Omega} \phi(\mathbf{x})(\lambda_1 e_1(\mathbf{x}) - \lambda_2 e_2(\mathbf{x}))d\mathbf{x} \end{aligned} \quad (2-9)$$

为了方便起见, 将上述能量泛函 $E^{GCRSF}(\phi)$ 简记为如下形式:

$$E^{GCRSF}(\phi) = |\nabla\phi|_g + \langle\phi, r\rangle \quad (2-10)$$

其中 $r(\mathbf{x}) = \lambda_1 e_1(\mathbf{x}) - \lambda_2 e_2(\mathbf{x})$, $\langle\phi, r\rangle = \int_{\Omega} \phi(\mathbf{x})r(\mathbf{x})d\mathbf{x}$.

于是, 新提出的 GCRSF 模型即是极小化式 (2-9) 或式 (2-10) 中能量泛函 $E^{GCRSF}(\phi)$, 即要解决的极小化问题就是:

$$\min_{\phi} E^{GCRSF}(\phi) = \min_{\phi} (|\nabla\phi|_g + \langle\phi, r\rangle) \quad (2-11)$$

注意到在式 (2-9) 或式 (2-10) 中定义的能量泛函 $E^{GCRSF}(\phi)$ 关于 ϕ 是一次齐次的, 因此一般情况下它没有唯一的全局极小点。这个问题是由曲线的水平集表示并不是唯一的引起的, 这可以通过将水平集函数 ϕ 限制在一个有限区间内, 如 $[a_0, b_0]$, 即是 $a_0 \leq \phi(\mathbf{x}) \leq b_0 (\forall \mathbf{x} \in \Omega)$ 来保证全局极小解的存在。 a_0 和 b_0 是两个常数。



2.2 全局凸区域可伸缩拟合能量模型



定义 2.4 新提出的 GCRSF 模型即是如下极小化问题：

$$\min_{a_0 \leq \phi \leq b_0} E^{GCRSF}(\phi) = \min_{a_0 \leq \phi \leq b_0} (|\nabla \phi|_g + \langle \phi, r \rangle) \quad (2-12)$$

一旦解出极小化问题 (2-12) 的解 ϕ ，可以通过对水平集函数 ϕ 设定一个阈值 $\alpha \in (a_0, b_0)$ ，将图像定义域 Ω 分割为两个互不相交的区域：

$$\begin{cases} \Omega_1 = \{\mathbf{x} : \phi(\mathbf{x}) > \alpha\} \\ \Omega_2 = \{\mathbf{x} : \phi(\mathbf{x}) \leq \alpha\} \end{cases} \quad (2-13)$$

阈值 α 一般取为 $\alpha = (a_0 + b_0)/2$ 。

事实上，在每一次迭代水平集函数 ϕ 之前，首先需要迭代拟合函数 $f_1(\mathbf{x})$ 和 $f_2(\mathbf{x})$ 。这里，与 RSF 模型一样，对于固定的水平集函数 ϕ ，可以给出两个拟合函数 $f_1(\mathbf{x})$ 和 $f_2(\mathbf{x})$ 的表达式如下：

$$f_i(\mathbf{x}) = \frac{K_\sigma(\mathbf{x}) * [M_i^\varepsilon(\phi(\mathbf{x}))u_0(\mathbf{x})]}{K_\sigma(\mathbf{x}) * M_i^\varepsilon(\phi(\mathbf{x}))} \quad (i = 1, 2) \quad (2-14)$$

其中 $K_\sigma(\mathbf{x})$ 是式 (1-23) 中定义的核函数。 $M_1^\varepsilon(\phi) = H_\varepsilon(\phi)$ ， $M_2^\varepsilon(\phi) = 1 - H_\varepsilon(\phi)$ 。 $*$ 是卷积算子： $a(\mathbf{x}) * b(\mathbf{x}) = \int a(\mathbf{x} - \mathbf{y})b(\mathbf{y})d\mathbf{y}$ 。



2.2 全局凸区域可伸缩拟合能量模型



定义 2.5 由 GCRSF 模型得到的原始图像 $u_0(\mathbf{x})$ 的拟合图像 $u(\mathbf{x})$ 可由拟合函数 $f_1(\mathbf{x})$ 和 $f_2(\mathbf{x})$ 定义如下:

$$u(\mathbf{x}) = \sum_{i=1}^2 f_i(\mathbf{x}) M_i^{\varepsilon}(\phi(\mathbf{x})) \quad (2-15)$$

由式(2-14)可知,新模型中计算 $f_1(\mathbf{x})$ 和 $f_2(\mathbf{x})$ 的表达式与 RSF 模型中计算 $f_1(\mathbf{x})$ 和 $f_2(\mathbf{x})$ 的表达式是一样的。但是对于固定的 $f_1(\mathbf{x})$ 和 $f_2(\mathbf{x})$, 关于水平集函数 ϕ 极小化能量泛函 $E^{GCRSF}(\phi)$ 时,不是用标准的梯度下降法,而是应用 Split Bregman 方法,详细的算法将在第 2.3 节中给出。

注解2.1 本节新提出的 GCRSF 模型不同于原始的 RSF 模型,相比 RSF 模型, GCRSF 模型有以下几点改进。

- 在式(2-9)或式(2-10)中新提出的能量泛函 $E^{GCRSF}(\phi)$ 不同于式(1-27)中 RSF 模型的能量泛函 $\mathcal{F}_{\varepsilon}^{RSF}(\phi, f_1, f_2)$ 。基于 RSF 模型的能量泛函关于水平集函数 ϕ 的梯度下降流方程,应用全局凸分割方法给出一个简化的梯度流方程,然后基于该简化的梯度流方程提出一个全局凸的能量泛函来保证全局极小解的存在。因此,新提出的能量泛函 $E^{GCRSF}(\phi)$ 可以看做是原 RSF 模型的能量泛函 $\mathcal{F}_{\varepsilon}^{RSF}(\phi, f_1, f_2)$ 的凸化和改进。



2.2 全局凸区域可伸缩拟合能量模型



- 根据 GAC 模型的思想, 为了更加容易地检测物体的边缘, 在能量泛函 $E^{GCRSF}(\phi)$ 中引入边界信息, 这是通过在能量泛函 $E^{GCRSF}(\phi)$ 中加入一个非负的边缘检测函数 g 来实现的, 这样, 新模型又可以得到与传统的蛇模型或者 GAC 模型类似的结果。于是, 新提出的 GCRSF 模型又可以看做是 GAC 模型与凸化以及简化的 RSF 模型的一个完美的结合。
- 新提出的 GCRSF 模型的能量泛函 $E^{GCRSF}(\phi)$ 的特殊结构, 即含有 L1 范数 $|\cdot|_1$ (这里用的是它的加权形式 $|\cdot|_g$), 使得在下一节中可以应用快速的 Split Bregman 方法来极小化能量泛函 $E^{GCRSF}(\phi)$ 。而 RSF 模型的能量泛函 $\mathcal{F}_\varepsilon^{RSF}(\phi, f_1, f_2)$ 并不具有这种特殊的结构, 因此极小化 RSF 能量泛函时, 只能使用标准的梯度下降法。Split Bregman 方法的应用保证了新提出的 GCRSF 模型比原始的 RSF 模型要快得多。



2.3 Split Bregman 方法极小化GCRSF 模型



由于在式(2-9) 或式(2-10) 中新提出的GCRSF 模型的能量泛函 $E_{GCRSF}(\phi)$ 含有L1 范数 $|\cdot|_1$ (这里用的是它的加权形式 $|\cdot|_g$) , 能量泛函 $E_{GCRSF}(\phi)$ 的特殊结构保证了可以应用第1.4 节中介绍的Split Bregman 方法来快速极小化它。

于是, 在本节中, 应用Split Bregman 方法来快速地解新提出的极小化问题(2-12):

$$\min_{a_0 \leq \phi \leq b_0} E_{GCRSF}(\phi) = \min_{a_0 \leq \phi \leq b_0} (|\nabla \phi|_g + \langle \phi, r \rangle) \quad (2-16)$$

根据Split Bregman 方法, 不是直接解极小化问题(2-16), 而是通过引入一个辅助变量 $\vec{d} = (d_x, d_y)$, 将(2-16) 转化为如下等价的约束极小化问题:

$$\min_{\substack{a_0 \leq \phi \leq b_0 \\ \vec{d}}} (|\vec{d}|_g + \langle \phi, r \rangle) \text{ 使得 } \vec{d} = \nabla \phi \quad (2-17)$$

为了解决约束极小化问题(2-17), 需要首先引入一个二次约束函数将它转化为如下无约束极小化问题:

$$(\phi^*, \vec{d}^*) = \arg \min_{\substack{a_0 \leq \phi \leq b_0 \\ \vec{d}}} (|\vec{d}|_g + \langle \phi, r \rangle + \frac{\lambda}{2} \|\vec{d} - \nabla \phi\|^2) \quad (2-18)$$

其中 λ 是一个正的常数



由第1.4节中介绍可知，式(2-18)中的二次约束函数 $\frac{\lambda}{2} \|\vec{d} - \nabla \phi\|^2$ 只是近似地或者称作弱地强约束条件 $\vec{d} = \nabla \phi$ 。

通过使用Bregman 迭代技巧来精确地或严格地强约束，对于与GCRSF模型极小化问题(2-16) 等价的约束极小化问题(2-17)，为了严格地强约束 $\vec{d} = \nabla \phi$ ，应用Bregman 迭代于(2-18)，在式(2-18)中的二次约束函数嵌入公式 $\frac{\lambda}{2} \|\vec{d} - \nabla \phi\|^2$ 中引入一个Bregman 变量 \vec{b} 在下面的定理2.5中给出解GCRSF 模型极小化问题(2-16) 的Split Bregman 迭代格式，并在定理2.6中给出该Split Bregman 迭代格式的收敛结果。

定理2.5 GCRSF 模型的极小化问题(2-16) 可以转化为如下一系列无约束优化问题：

$$(\phi^{k+1}, \vec{d}^{k+1}) = \arg \min_{\substack{\vec{d} \\ a_0 \leq \phi \leq b_0}} \left(|\vec{d}|_g + \langle \phi, r^k \rangle + \frac{\lambda}{2} \|\vec{d} - \nabla \phi - \vec{b}^k\|^2 \right) \quad (2-29)$$

其中 $\vec{b} = (b_x, b_y)$ 是引入的Bregman 变量， \vec{b}^{k+1} 可由如下Bregman 迭代得出：

$$\vec{b}^{k+1} = \vec{b}^k + (\nabla \phi^{k+1} - \vec{d}^{k+1}), \quad \vec{b}^0 = \mathbf{0} = (0, 0) \quad (2-30)$$



上述式(2-29) 和式(2-30) 即是关于水平集函数 ϕ 极小化GCRSF 模型能量泛函 $E^{GCRSF}(\phi)$ 的Split Bregman 迭代算法。对于该Split Bregman 迭代算法，有如下收敛结果。

定理2.6 极小化GCRSF 模型能量泛函的Split Bregman 迭代算法式(2-29) 和式(2-30) 在如下意义下收敛：当 $k \rightarrow \infty$ 时， $\|\vec{d}^k - \nabla \phi^k\| \rightarrow 0$ 与 $\|\phi^k - \phi^*\| \rightarrow 0$ ，其中 ϕ^* 是极小化问题(2-16) 的解。即一系列无约束优化问题(2-29) 的解 ϕ^k 收敛于原始的极小化问题(2-16) 的解 ϕ^* 。

注解2.2 Split Bregman 方法或Bregman 迭代技巧相对于传统的连续方法的几个优点已经在第1.4 节中详细说明，如关于 ϕ 极小化(2-29) 时不需要传统的正则化过程，参数 λ 可以取作常数从而保证了迭代优化方法的快速收敛，避免了连续方法中 $\lambda \rightarrow \infty$ 时的数值不稳定问题等，这里不再详述。

为了解无约束优化问题(2-29)，应用第1.4 节中介绍的IM 格式。即分别关于 ϕ 和 \vec{d} 交替极小化(2-29)，这可以通过以下两步完成：

- 第一步：对于固定的 \vec{d}^k ，关于 ϕ 极小化(2-29)：

$$\phi^{k+1} = \arg \min_{a_0 \leq \phi \leq b_0} \left(\langle \phi, r^k \rangle + \frac{\lambda}{2} \|\vec{d}^k - \nabla \phi - \vec{b}^k\|^2 \right) \quad (2-34)$$

- 第二步：对于固定的 ϕ^{k+1} ，关于 \vec{d} 极小化(2-29)：

$$\vec{d}^{k+1} = \arg \min_{\vec{d}} \left(|\vec{d}|_g + \frac{\lambda}{2} \|\vec{d} - \nabla \phi^{k+1} - \vec{b}^k\|^2 \right) \quad (2-35)$$

对于第一步，由于已经用Split Bregman 方法将 ϕ 分离出来，关于 ϕ 的优化问题现在是可微的。这里应用变分方法和Gauss-Seidel 方法解关于 ϕ 的极小化问题(2-34)，在如下定理2.7 中给出 ϕ^{k+1} 的迭代格式。

定理2.7 对于固定的 \vec{d}^k ，极小化问题(2-34) 的极小点 ϕ^{k+1} 满足如下欧拉-拉格朗日方程：

$$\Delta \phi^{k+1} = \frac{1}{\lambda} r^k + \nabla \cdot (\vec{d}^k - \vec{b}^k) \quad (a_0 \leq \phi^{k+1} \leq b_0) \quad (2-36)$$



对于该拉普拉斯方程，应用Gauss-Seidel 思想求解，则对于图像中的每一个像素点 (i, j) ， $\phi_{i,j}^{k+1}$ 可由如下格式迭代：

$$\begin{cases} \alpha_{i,j}^k = d_{x,i-1,j}^k - d_{x,i,j}^k + d_{y,i,j-1}^k - d_{y,i,j}^k - (b_{x,i-1,j}^k - b_{x,i,j}^k + b_{y,i,j-1}^k - b_{y,i,j}^k) \\ \beta_{i,j}^k = \frac{1}{4}(\phi_{i-1,j}^k + \phi_{i+1,j}^k + \phi_{i,j-1}^k + \phi_{i,j+1}^k - \frac{1}{\lambda}r_{i,j}^k + \alpha_{i,j}^k) \\ \phi_{i,j}^{k+1} = \max\{\min\{\beta_{i,j}^k, b_0\}, a_0\} \end{cases} \quad (2-37)$$

证明：对于固定的 \vec{d}^k ，关于 ϕ 解极小化问题(2-34)，记

$$E_1(\phi) = \langle \phi, r^k \rangle + \frac{\lambda}{2} \|\vec{d}^k - \nabla \phi - \vec{b}^k\|^2$$

现在 $E_1(\phi)$ 关于 ϕ 是可微的，因此 $E_1(\phi)$ 的极小点 ϕ^{k+1} 满足：

$$E'_1(\phi^{k+1}) = 0$$

可以很容易地推导出：

$$E'_1(\phi) = r^k - \lambda \Delta \phi + \lambda \nabla \cdot (\vec{d}^k - \vec{b}^k)$$

于是，极小点 ϕ^{k+1} 满足：

$$r^k - \lambda \Delta \phi^{k+1} + \lambda \nabla \cdot (\vec{d}^k - \vec{b}^k) = 0$$

进而满足如下拉普拉斯方程：

$$\Delta \phi^{k+1} = \frac{1}{\lambda} r^k + \nabla \cdot (\vec{d}^k - \vec{b}^k)$$

注意这里需要将 ϕ^{k+1} 限制在有限区间 $[a_0, b_0]$ 里，即可得(2-36)式。

为了解关于 ϕ 的拉普拉斯方程(2-36)，在实际中需要将其离散化：

$$\Delta \phi_{i,j}^{k+1} = \frac{1}{\lambda} r_{i,j}^k + \nabla \cdot (\vec{d}^k - \vec{b}^k)_{i,j}$$

对于拉普拉斯算子和散度算子，分别采用中心差分法和向后差分法，并应用 Gauss-Seidel 方法的思想，这样对于图像中的每一个像素点 (i, j) ，有：

$$\Delta \phi_{i,j}^{k+1} = \phi_{i-1,j}^k + \phi_{i+1,j}^k + \phi_{i,j-1}^k + \phi_{i,j+1}^k - 4\phi_{i,j}^{k+1}$$

$$\begin{aligned} \nabla \cdot (\vec{d}^k - \vec{b}^k)_{i,j} &= (\nabla_x, \nabla_y) \cdot (d_x^k - b_x^k, d_y^k - b_y^k)_{i,j} \\ &= \nabla_x (d_x^k - b_x^k)_{i,j} + \nabla_y (d_y^k - b_y^k)_{i,j} \\ &= \nabla_x d_{x,i,j}^k + \nabla_y d_{y,i,j}^k - (\nabla_x b_{x,i,j}^k + \nabla_y b_{y,i,j}^k) \\ &= d_{x,i,j}^k - d_{x,i-1,j}^k + d_{y,i,j}^k - d_{y,i,j-1}^k - (b_{x,i,j}^k - b_{x,i-1,j}^k + b_{y,i,j}^k - b_{y,i,j-1}^k) \end{aligned}$$



将上面两个式子带入离散化的拉普拉斯方程，记

$$\alpha_{i,j}^k = d_{x,i-1,j}^k - d_{x,i,j}^k + d_{y,i,j-1}^k - d_{y,i,j}^k - (b_{x,i-1,j}^k - b_{x,i,j}^k + b_{y,i,j-1}^k - b_{y,i,j}^k)$$

可以得到：

$$\phi_{i,j}^{k+1} = \frac{1}{4}(\phi_{i-1,j}^k + \phi_{i+1,j}^k + \phi_{i,j-1}^k + \phi_{i,j+1}^k - \frac{1}{\lambda}r_{i,j}^k + \alpha_{i,j}^k)$$

考虑到每一次迭代中，都需要将 $\phi_{i,j}^{k+1}$ 限制在有限区间 $[a_0, b_0]$ 里，记

$$\beta_{i,j}^k = \frac{1}{4}(\phi_{i-1,j}^k + \phi_{i+1,j}^k + \phi_{i,j-1}^k + \phi_{i,j+1}^k - \frac{1}{\lambda}r_{i,j}^k + \alpha_{i,j}^k)$$

则可得：

$$\phi_{i,j}^{k+1} = \max\{\min\{\beta_{i,j}^k, b_0\}, a_0\}$$

因此，可以得到对于图像中的每一个像素点 (i, j) ， $\phi_{i,j}^{k+1}$ 的迭代格式即是式(2-37)。□

对于第二步, 关于 \vec{d} 极小化(2-35), 可以利用向量值shrinkage 算子显式地计算出 \vec{d}^{k+1} 。

定理2.8 对于固定的 ϕ^{k+1} , 极小化问题(2-35) 的极小点 \vec{d}^{k+1} 可以由向量值shrinkage 算子给出如下显式的迭代格式:

$$\vec{d}^{k+1} = \text{shrink}_g \left(\vec{b}^k + \nabla \phi^{k+1}, \frac{1}{\lambda} \right) = \text{shrink} \left(\vec{b}^k + \nabla \phi^{k+1}, \frac{g}{\lambda} \right) \quad (2-38)$$

其中 $\text{shrink}_g(x, \gamma)$ 是加权的向量值shrinkage 算子, 向量值shrinkage 算子 $\text{shrink}(x, \gamma)$ 的定义见定义1.3。

事实上, 由于 $\vec{d} = (d_x, d_y)$, $\nabla \phi = (\nabla_x \phi, \nabla_y \phi)$, $\vec{b} = (b_x, b_y)$, 因此也可以直接将式(2-38) 写为如下形式:

$$\begin{cases} d_x^{k+1} = \max \left(s^k - \frac{g}{\lambda}, 0 \right) \frac{\nabla_x \phi^{k+1} + b_x^k}{s^k} \\ d_y^{k+1} = \max \left(s^k - \frac{g}{\lambda}, 0 \right) \frac{\nabla_y \phi^{k+1} + b_y^k}{s^k} \end{cases} \quad (s^k \neq 0) \quad (2-39)$$

其中

$$s^k = \sqrt{|\nabla_x \phi^{k+1} + b_x^k|^2 + |\nabla_y \phi^{k+1} + b_y^k|^2} \quad (2-40)$$



若 $s^k = 0$, 则 $d_x^{k+1} = d_y^{k+1} = 0$ 。这里需要说明的是, 由式(2-39) 迭代 d_x^{k+1} 与 d_y^{k+1} 时, 都是在每一个像素点 (i, j) 上进行的。对于 $(\nabla_x \phi^{k+1})_{i,j}$ 和 $(\nabla_y \phi^{k+1})_{i,j}$, 采用如下差分格式:

$$(\nabla_x \phi^{k+1})_{i,j} = \phi_{i+1,j}^{k+1} - \phi_{i,j}^{k+1}$$

$$(\nabla_y \phi^{k+1})_{i,j} = \phi_{i,j+1}^{k+1} - \phi_{i,j}^{k+1}$$

因此, 上述shrinkage 算法是很容易实现的并且是非常快速的。

极小化 GCRSF 模型 (2-16) 的 Split Bregman 算法:

```
Input:  $u_0, \phi_0$ , and  $\vec{d}^0 = \vec{b}^0 = \mathbf{0} = (0, 0)$   
1 Compute initial  $f_1^0$  and  $f_2^0$   
2 while  $\|\phi^{k+1} - \phi^k\| > \varepsilon$  do  
3   Define  $r^k = \lambda_1 e_1^k - \lambda_2 e_2^k$   
4    $\phi^{k+1} = GS(r^k, \vec{d}^k, \vec{b}^k, \lambda)$   
5    $\vec{d}^{k+1} = \text{shrink}_g(\vec{b}^k + \nabla \phi^{k+1}, \frac{1}{\lambda})$   
6    $\vec{b}^{k+1} = \vec{b}^k + (\nabla \phi^{k+1} - \vec{d}^{k+1})$   
7   Find  $C^k = \{\mathbf{x} : \phi^k(\mathbf{x}) = \alpha\}$   
8   Find  $u^k = \sum_{i=1}^2 f_i^k M_i^\varepsilon(\phi^k)$   
9   Update  $f_1^k$  and  $f_2^k$   
10 end  
Output:  $C^k$  and  $u^k$ 
```

- $GS(r^k, \vec{d}^k, \vec{b}^k, \lambda)$ 来表示 Gauss Seidel 格式 (2-37)。
- C^k 和 u^k 分别是活动轮廓线和拟合图像。



2.4.1 数值实现与实验结果



将水平集函数 ϕ 简单地初始化为 ϕ_0 ，其中 ϕ_0 是一个二值阶梯函数，在定义域的某一个区域内部（包含边界）取值为 b_0 ，外部则取值为 a_0 。除非特殊说明，选择 $a_0 = -2$ ， $b_0 = 2$ ，这样阈值 $\alpha = (a_0 + b_0) = 0$ 就被用来寻找要分割的物体或者边界。

用二值函数作为初始的水平集函数有两个优点。

- 第一，新的轮廓很容易出现；
- 第二，曲线演变的速度比用符号距离函数作为初始水平集函数要快得多。这是因为在水平集演变过程中，并不需要将水平集函数重新初始化为它的零水平集的符号距离函数这一过程。只需要将水平集函数 ϕ 限制在有限区间 $[a_0, b_0]$ 内即可，这样，水平集函数 ϕ 的绝对值在它的零水平集两侧都不会变得过大，从而也不会引起不准确的计算或者是错误的分割结果。同时又省去了重新初始化这一繁琐耗时的过程，提高了演变的速度。对于光滑的 Heaviside 函数 H_ε ，与 Chan 等人，Li 等人以及 Wang 等人一样，取 $\varepsilon = 1$ 。



2.4.1 数值实现与实验结果



在算法的实现中，每一次更新水平集函数 ϕ 之前，都要首先由式(2-14)更新两个拟合函数 $f_1(\mathbf{x})$ 和 $f_2(\mathbf{x})$ 。由式(2-14)可知，要计算 $f_1(\mathbf{x})$ 和 $f_2(\mathbf{x})$ ，需要计算卷积。与 RSF 模型类似，这里为了更加快速地计算式(2-14)中的卷积，将核函数 K_σ 截断为一个 $n \times n$ 的掩膜，这里 n 是不小于 4σ 的最小奇数。

例如：

给定一个尺度参数 $\sigma = 3.0$ ，有 $n = 13$ ，即掩膜的大小为 13×13 。用一个大小为 13×13 的掩膜与用一个较大的掩膜，如 41×41 ($\sigma = 10.0$) 的区别，对于图像分割的结果而言是可以忽略的，但是使用较大的掩膜会使得计算量增大。因此，为了更有效地计算卷积，取 $\sigma = 3.0$ 是适当的。



2.4.1 数值实现与实验结果



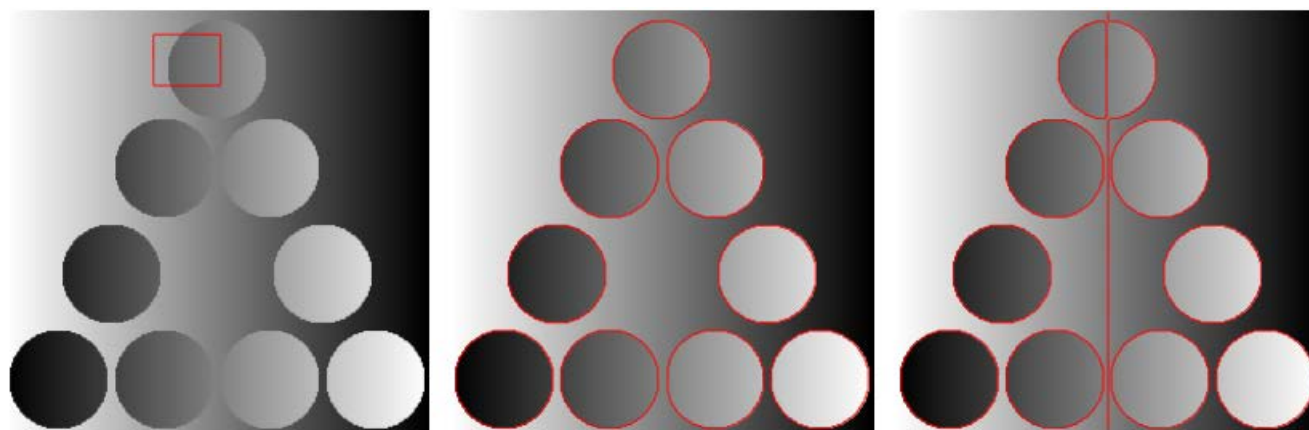
λ 是式(2-29)中二次函数前的参数, 根据第2.3节中的介绍, Split Bregman方法允许使用一个固定的常数 λ , 对于本章提出的 GCRSF 模型, 使用 $\lambda = 0.001$ 。 $\beta \geq 0$ 是边缘检测函数(2-8)中的参数, 它可以用来决定分割的细节水平。

- 如果想要检测更多细节, 应该使用较大的 β ;
- 反之, 如果只想给出物体的轮廓, 就应该使用较小的 β 。

对于本章在下一节中用到的所有实验图像, 除非特殊说明, 使用:

- $\beta = 100$ 。
- 参数 λ_1 和 λ_2 分别是轮廓线 C 内部区域 $\Omega_1 = \text{inside}(C)$ 和外部区域 $\Omega_2 = \text{outside}(C)$ 的权系数。在大多数情况下, 将 λ_1 和 λ_2 取为相同的常数, 从而引导零水平轮廓线 C 内部区域与外部区域之间的公平竞争。但是, 当使用不同的常数 λ_1 和 λ_2 时, 强加于内部区域与外部区域上的惩戒力就不同。例如, 当 $\lambda_1 > \lambda_2$ 时, 强加于内部区域上的惩戒力就较大, 这也意味着较大的惩戒力作用于内部区域 $\text{inside}(C)$ 的面积, 这就在一定程度上阻止了初始轮廓线外部新轮廓线的出现。这是因为在初始轮廓线外部出现新的轮廓线将会增大内部区域 $\text{inside}(C)$ 的面积。在本章中, 除了特殊说明外, 取 $\lambda_1 = \lambda_2 = 1e - 5$ 。

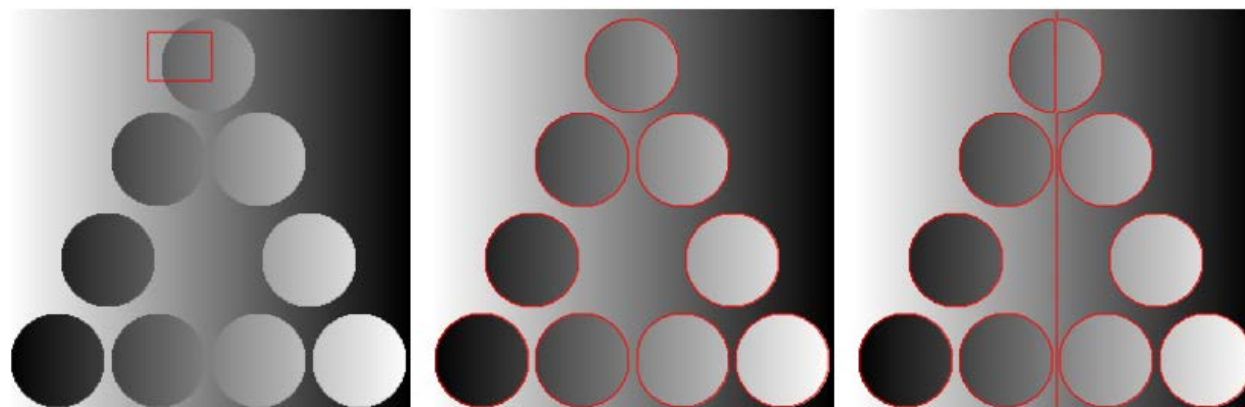
首先将 GCRSF 模型应用于一个具有图像强度不均匀性质的合成图像。该图像的大小为 256×256 。图2-1给出了用 GCRSF 模型与 Goldstein 等人的全局凸 CV 模型分割该图像得到的不同结果之间的比较。



a) 原始图像与初始轮廓线 b) GCRSF 模型的结果 c) 全局凸 CV 模型的结果
a) The original image with the initial contour b) The result of the GCRSF model c) The result of the globally convex CV model

图 2-1 GCRSF 模型与全局凸 CV 模型的比较

由图2-1a)中的原始图像可知，该图像无论是前景（物体）还是背景都存在严重的图像强度不均匀性质。并且图2-1a)中给出的相同的初始轮廓线；



a) 原始图像与初始轮廓线 b) GCRSF 模型的结果 c) 全局凸 CV 模型的结果
a) The original image with the initial contour b) The result of the GCRSF model c) The result of the globally convex CV model

图 2-1 GCRSF 模型与全局凸 CV 模型的比较

- 图2-1b)和图2-1c)分别是GCRSF模型与全局凸CV模型的分割结果。由于全局凸 CV 模型仅仅考虑图像的全局强度信息，主要适用于具有同质区域的图像，因此它不能很好地分割这一个强度严重不均匀的图像。
- 由图2-1c)中给出的结果明显地看出，全局凸 CV 模型将背景错误地分割为两部分。而 GCRSF 模型充分利用了图像的局部强度信息，因此可以很好地处理该图像，将物体精确地分割出来的同时，还保留了完整的背景，如图2-1b)所示。这个实验很好地显示了 GCRSF 模型相比全局凸CV模型在分割具有强度不均匀性质图像上的优越性。

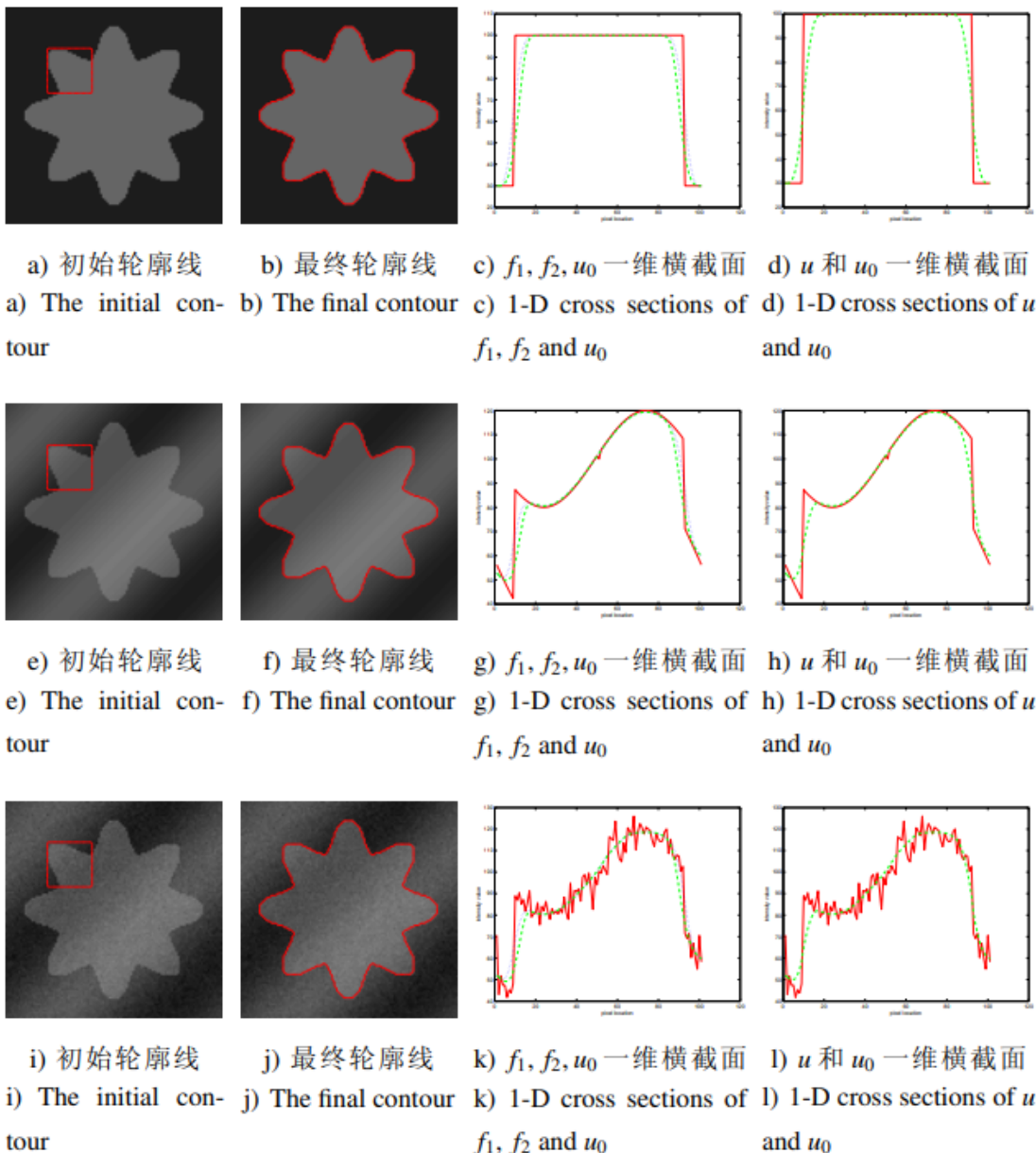


图2-2给出了 GCRSF 模型应用于三个合成的花瓣形图像的结果。这三个图像有着不同的图像强度分布。图像的大小均为 101×101 。这三个原始图像以及初始轮廓线如图2-2a), 图2-2e) 和图2-2i) 所示。其中

- 图 2-2a) 所示是分段常量图像;
- 图 2-2e) 所示是强度不均匀的图像;
- 图2-2i) 所示是在第二个图像2-2e) 中加入标准差为 5.0 的随机噪声生成的。
- 图2-2b), 图2-2f) 和图2-2j) 是 GCRSF 模型的分割结果。由此可以看出, GCRSF 模型可以很好地处理这三个图像, 即使图像强度不均匀或者强度不均匀并且含有噪声;

图 2-2 GCRSF 模型用于三个合成图像的结果



2.4.2 实验结果



CV 模型和全局凸 CV 模型仅仅可以分割分段常量的图像，不能处理强度不均匀的图像。由于本实验中的第三个图像图2-2i) 含有较高水平的噪声，因此对于图2-2i)，取较小的 $\beta = 20$ 。

为了更加清楚地看到与最终水平集函数 ϕ 相关的两个拟合函数 f_1 , f_2 以及拟合图像 $u = \sum_{i=1}^2 M_i^\varepsilon(\phi) f_i$ 的值，在第三列和第四列中给出 f_1, f_2, u 和原始图像 u_0 在中间行的一维横截面图。其中横轴表示像素位置，纵轴表示图像强度值。

- 第三列给出 f_1 （蓝色点线）， f_2 （绿色虚线）和原始图像 u_0 （红色实线）的一维横截面图，可以看出 f_1 和 f_2 都是光滑的，即使是在有比较大的噪声存在的情况下（图2-2k)）。事实上， f_1 和 f_2 的光滑性可以由它们的表达式(2-14)中卷积的性质直接得出。
- 第四列给出拟合图像 u （绿色虚线）和原始图像 u_0 （红色实线）的一维横截面图，可以看出拟合图像 u 可以很好地近似原始图像 u_0 。此外，通过比较图2-2g) 和图2-2k)，以及图2-2h) 和图2-2l)，可以看到，即使存在比较大的噪声，用GCRSF 模型得到的最终的 f_1, f_2, u 仅仅很轻微地受到噪声的影响，含噪声或者不含噪声的分割结果几乎是一样的，这也证明了GCRSF 模型对噪声的鲁棒性。

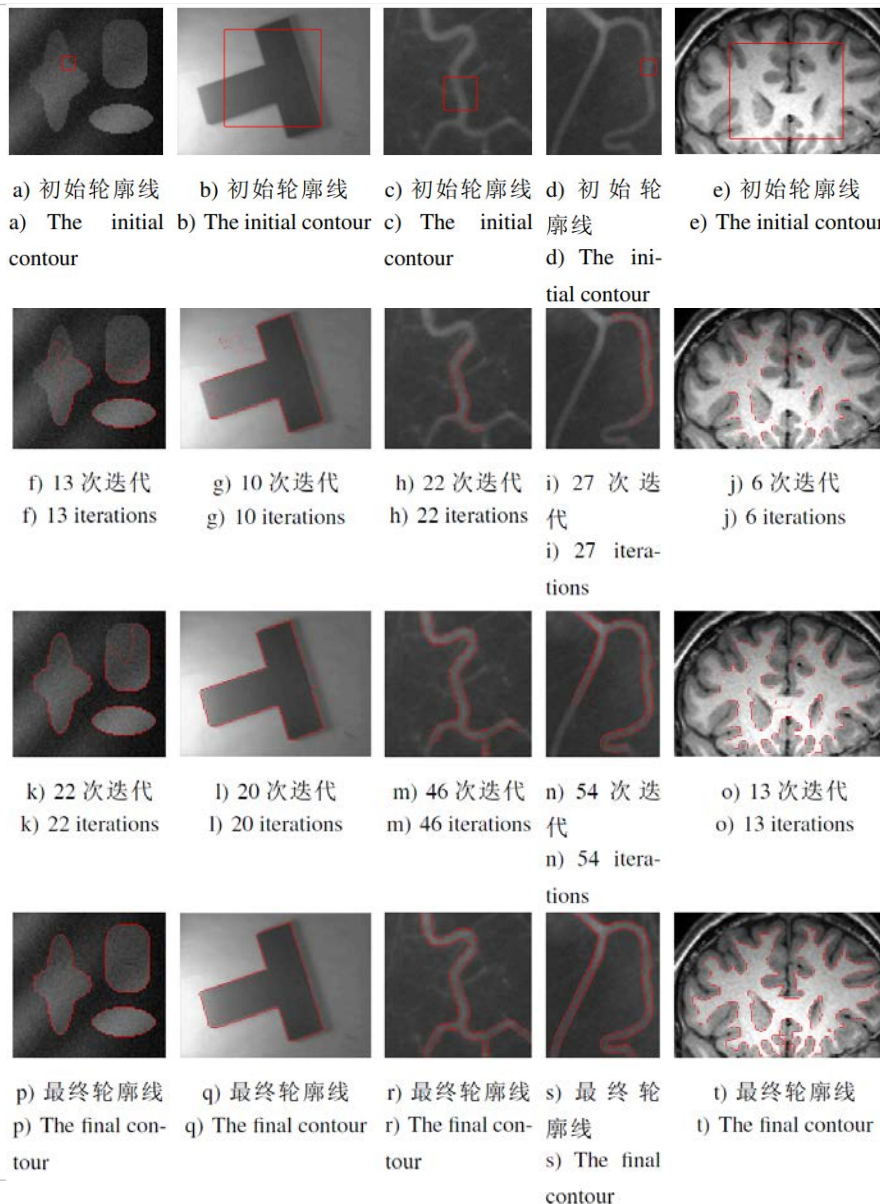


2.4.2 实验结果



将 GCRSF 模型应用于几个合成和真实的强度不均匀图像，分割结果以及轮廓线演变过程见图2-3。

- 图2-3a), 图2-3b), 图2-3c), 图2-3d) 和图2-3e) 原始图像与初始轮廓线; 这几个图像的大小依次为 75×79 , 96×127 , 110×111 , 131×103 和 78×119 。
- 这几个图像都是典型的具有强度不均匀性质的图像，从左到右依次是一个合成的图像，一个真实的T形物体图像，两个血管的 X 射线图像和一个大脑 MR 图像。
- 图2-3a) 中的合成图像强度分布很不均匀而且含有很大的噪声；图2-3b) 中的 T 形物体图像由于光照的原因图像强度也是不均匀的；图2-3c) 和图2-3d) 中的两个血管图像不仅强度分布不均匀，而且部分血管的边缘非常弱的；另外，对于图2-3e) 中的大脑 MR 图像，可以看到上面的部分白质的图像强度甚至比下面的部分灰质的图像强度还要低。





2.4.2 实验结果



将 GCRSF 模型应用于这几个图像，并在每一列给出相应轮廓线演变过程。由最后一行给出最终轮廓线，可以看出 GCRSF 模型可以成功地提取物体边缘，将物体很好地分割出来。这是因为 GCRSF 模型能充分利用图像的局部强度信息，从而处理严重的图像强度不均匀性与弱边缘的问题。在该实验中，对于图 2-3d) 和图 2-3e)，分别取 $\lambda_1 = 3.1e - 5$, $\lambda_2 = 1e - 5$ 和 $\lambda_1 = 1.25e - 5$, $\lambda_2 = 1e - 5$ 。取 $\lambda_1 > \lambda_2$ ，增大强加于内部区域上的惩戒力，从而阻止了轮廓线外部新轮廓线的出现。

由于图 2-3a) 含有噪声，取 $\beta = 20$ 。对于图 2-3e)，为了避免多余的细节出现，也取较小的 $\beta = 10$ 。



2.4.2 实验结果



事实上，Li 等人中已经应用 RSF 模型测试过这几个图像，并得到类似的分割结果，但是通过表2-1和表2-2中给出的用这两个模型处理图像的迭代次数和计算时间的比较，可以看出，GCRSF 模型比 RSF模型要快得多。GCRSF 模型可以快速地分割给定的图像，是由以下几点保证的。

GCRSF 能量泛函是凸的，这就保证了算法的快速收敛。

其次，不需要重新初始化水平集函数 ϕ 为它的零水平集的符号距离函数，只需要将水平集函数 ϕ 简单地初始化为一个二值阶梯函数，并且在演变过程中将 ϕ 限制在有限区间 $[a_0, b_0]$ 内即可，这样新的轮廓线很容易出现，而且曲线演变的速度也很快。

最后，也是最重要的，可以应用快速的 Split Bregman 算法来极小化能量泛函 $E^{GCRSF}(\phi)$ 。



2.4.2 实验结果



表 2-1 GCRSF 模型与 RSF 模型迭代次数的比较

Table2-1 Comparison of the iteration number of the GCRSF model and the RSF model

	图 2-3 a)	图 2-3 b)	图 2-3 c)	图 2-3 d)	图 2-3 e)
GCRSF 模型	32	26	67	82	48
RSF 模型	200	300	150	220	300

表 2-2 GCRSF 模型与 RSF 模型计算时间（秒）的比较

Table2-2 Comparison of the CPU time (second) of the GCRSF model and the RSF model

	图 2-3 a)	图 2-3 b)	图 2-3 c)	图 2-3 d)	图 2-3 e)
GCRSF 模型	0.33	0.49	1.13	1.20	0.70
RSF 模型	1.40	3.72	1.74	3.42	3.01

这里实验的迭代次数和计算时间是在Intel(R) Core(TM)2 Quad CPU Q8200, 2.33GHz, 2GB RAM 的ACPI 多处理器个人计算机, Windows XP 系统下的MATLAB R2010a 上运行Matlab 程式记录得到的。

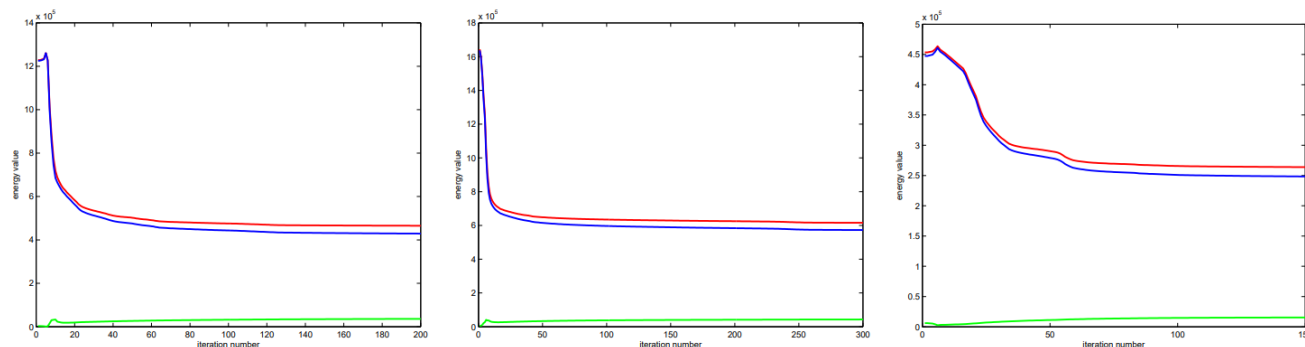


2.4.2 实验结果

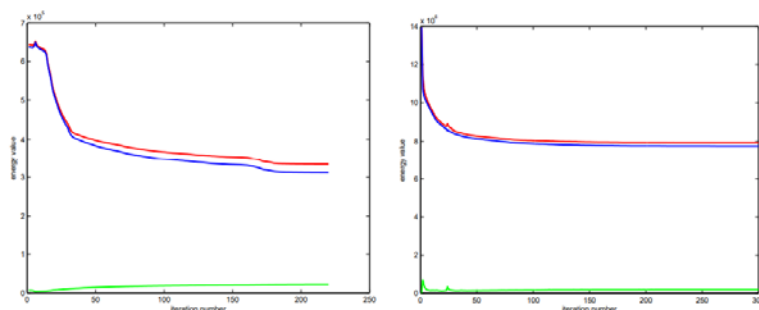


在提出 GCRSF 模型的过程中，去掉了原 RSF 模型梯度下降流方程(2-1)中的第三项 $\mu \left(\nabla^2 \phi - \operatorname{div} \left(\frac{\nabla \phi}{|\nabla \phi|} \right) \right)$ ，这一项对应于 RSF 模型能量泛函(1-25) 中的 $\mu \mathcal{P}(\phi)$ ，下面以图 2-3 中的五个图像为例，说明去掉这一项是合理性。

- 对于图 2-3 中的每一个图像，给出原来的带有 $\mu \mathcal{P}(\phi)$ 的能量（红线），去掉 $\mu \mathcal{P}(\phi)$ 后的能量（蓝线）和 $\mu \mathcal{P}(\phi)$ 这一项自身的能量（绿线）三者大小的比较，如图 2-4 所示。
- 对于图 2-4 中的各个子图，横轴表示迭代次数，纵轴表示能量的大小。可以明显地看出，带不带最后一项 $\mu \mathcal{P}(\phi)$ 的能量变化形状几乎是一样的，而且 $\mu \mathcal{P}(\phi)$ 自身的大小相对于总能量的大小是非常小的，因此可以被忽略。
- 图 2-3 中的结果也显示去掉这一项并不会影响 GCRSF 模型的效果，而且去掉该项保证了 GCS 方法和 Split Bregman 方法的应用，从而使得新提出的 GCRSF 模型比原来的 RSF 模型快得多，这可以由表 2-1 和表 2-2 看出。



a) 对应于图 2-3 a) 中的图像 b) 对应于图 2-3 b) 中的图像 c) 对应于图 2-3 d) 中的图像
a) For the image in Fig. 2-3 a) b) For the image in Fig. 2-3 b) c) For the image in Fig. 2-3 c)



d) 对应于图 2-3 d) 中的图像 e) 对应于图 2-3 e) 中的图像
d) For the image in Fig. 2-3 d) e) For the image in Fig. 2-3 e)

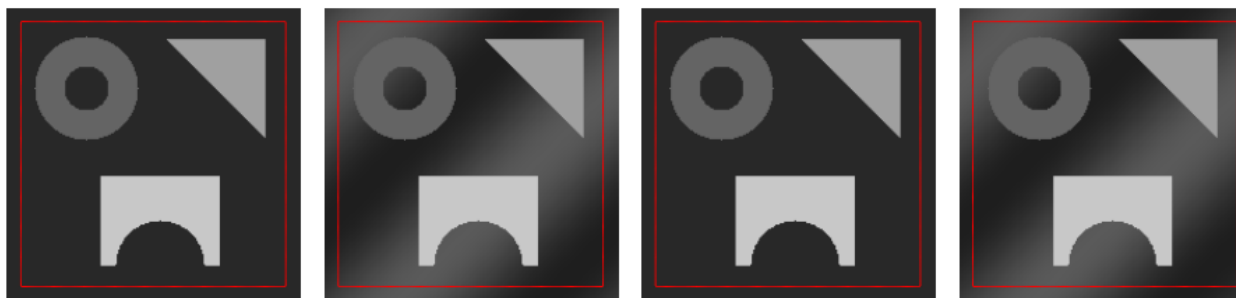
图 2-4 带与不带 $\mu P(\phi)$ 的能量变化的比较



2.4.2 实验结果



图 2-5 给出了 GCRSF 模型应用于两个合成图像的结果。这两个合成图像如图2-5a)和图2-5b)所示,它们都有三个具有不同强度值的物体,图像大小均为 200×200 。不同之处在于,图2-5a)中背景强度值是一个常量,而图 2-5b)中背景强度是不均匀的。对于这两个合成图像,如果取 $a_0 = -2$ 和 $b_0 = 2$,并且用图2-5a)和图2-5b)中所示的初始轮廓线,则利用 GCRSF 模型得到相应的分割结果见图2-5e)和图2-5f)。可以看出, GCRSF 模型可以准确地提取三个物体的边缘,即使在图2-5b)中,左上角背景处的图像强度值与圆环的强度值非常接近。但是如果取 $a_0 = 0$ 和 $b_0 = 1$,并用相同的初始轮廓线如图2-5c)和图2-5d)所示,得到相应的最终轮廓线见图2-5g)和图2-5h),圆环内部的部分边界将会丢失。在该实验中,对于强度不均匀的图 2-5 b)或者图 2-5 d),取 $\lambda_1 = \lambda_2 = 2e - 6$ 。这个实验说明了 a_0 和 b_0 的取值可以影响到 GCRSF 模型的结果,一般情况下,取 $a_0 = -2$ 和 $b_0 = 2$,都能得到比较满意的结果。此外,这个例子还说明 GCRSF 模型可以分割具有多个不同强度均值的图像。



a) 初始轮廓线

b) 初始轮廓线

c) 初始轮廓线

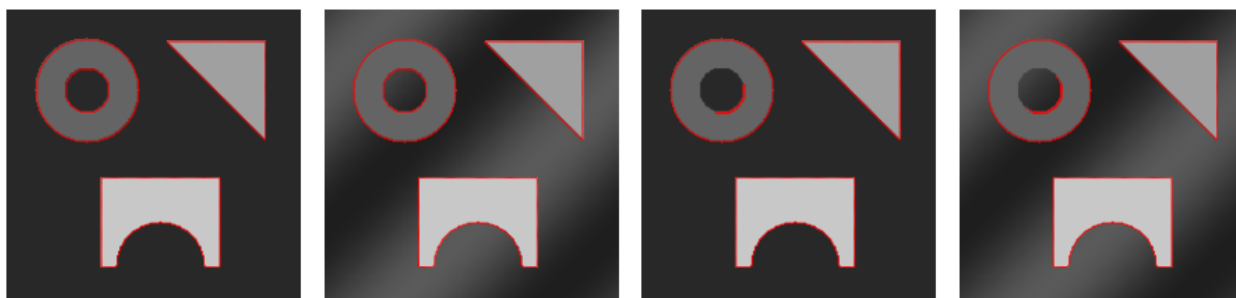
d) 初始轮廓线

a) The initial contour

b) The initial contour

c) The initial contour

d) The initial contour



e) 最终轮廓线

f) 最终轮廓线

g) 最终轮廓线

h) 最终轮廓线

e) The final contour

f) The final contour

g) The final contour

h) The final contour

图 2-5 a_0 和 b_0 对 GCRSF 模型用于两个合成图像的影响



2.4.2 实验结果



GCRSF 模型可以很容易地被推广到彩色图像。图 2-6 给出了 GCRSF 模型应用于一个真实的彩色花瓣图像的结果。该图像的大小为 450×449 。

图2-6的第一行给出了活动轮廓线从初始轮廓线到最终轮廓线的演变过程。由图2-6d)中给出的最终轮廓线，可以看出，虽然这个彩色图像的背景强度十分不均匀，但是 GCRSF 模型可以正确地将彩色花瓣的边界提取出来。

在第二行和第三行中给出对应的两个拟合函数 f_1 , f_2 在不同迭代次数的演变过程。可以看出，这两行中给出的图像是光滑的，这也进一步从实验上验证了拟合函数 f_1 , f_2 的光滑性。

第四行给出 f_1 , f_2 的组合 $u = \sum_{i=1}^2 M_i^{\varepsilon}(\phi) f_i$ ，即是拟合图像的演变过程。

对于该图像，取 $\lambda_1 = \lambda_2 = 2e - 6$ 。这个例子说明 GCRSF模型可以很好地处理具有图像强度不均匀性质的彩色图像。



a) 初始轮廓线 b) 240 次迭代 c) 340 次迭代 d) 最终轮廓线
a) The initial contour b) 240 iterations c) 340 iterations d) The final contour



e) 初始 f_1 f) 240 次迭代 g) 340 次迭代 h) 最终 f_1
e) The initial f_1 f) 240 iterations g) 340 iterations h) The final f_1



i) 初始 f_2 j) 240 次迭代 k) 340 次迭代 l) 最终 f_2
i) The initial f_2 j) 240 iterations k) 340 iterations l) The final f_2



m) 初始 u n) 240 次迭代 o) 340 次迭代 p) 最终 u
m) The initial u n) 240 iterations o) 340 iterations p) The final u

图 2-6 GCRSF 模型应用于一个真实的彩色花瓣图像的结果