

# 基于改进的最大类间方差算法的图像分割研究

瞿 中

(重庆邮电大学计算机科学与技术学院 重庆 400065) (重庆大学计算机学院 重庆 400030)

**摘要** 每种图像分割方法都只利用了图像信息中的部分特征,必然带有局限性,因此只能针对各种实际应用领域的需求来适当选择所需的方法。比较了几种阈值分割和边缘检测算法,着重研究了最大类间方差算法,并对其进行改进。针对不同的图像进行了仿真,对实验结果进行了分析、研究、比较。结果表明,改进的 Otsu 算法能有效地提高图像分割的质量。

**关键词** 最大类间方差算法,图像分割,阈值分割,边缘检测

**中图分类号** TP391.41 **文献标识码** A

## Research on Image Segmentation Based on the Improved Otsu Algorithm

QU Zhong

(College of Computer Science & Technology, Chongqing University of Posts & Telecommunications, Chongqing 400065, China)

(College of Computer Science, Chongqing University, Chongqing 400030, China)

**Abstract** Some characteristics were used in the image segmentation algorithms, these algorithms have the inevitable limitations, so the appropriate methods were selected for a variety of practical applications in the field of demand. Some threshold segmentation, edge detection algorithms were discussed, the Otsu algorithm and the improved Otsu algorithm were studied. The results of the simulation by different images were analysed, studied, compared. The results show that the improved Otsu algorithm can effectively improve the quality of image segmentation.

**Keywords** Otsu algorithm, Image segmentation, Threshold segmentation, Edge detection

在对图像的研究和应用中,人们往往仅对图像中的某些部分感兴趣,这些部分常称为目标或对象,它们一般对应图像中特定的、具有独特性质的区域。图像分割就是指把图像分成各具特性的区域,并利用图像信息中部分特征提取图像中若干感兴趣目标的技术和过程<sup>[1,2]</sup>。图像分割是图像处理的关键环节,又是一个经典难题,自 20 世纪 70 年代以来一直吸引着国内外研究者为之努力,至今已提出上千种分割算法。

各种图像分割方法都只利用了图像信息中的部分特征,必然带有局限性,因此只能针对各种实际应用领域的需求来适当选择所需的方法。本文比较了几种阈值分割、边缘检测算法,着重研究了最大类间方差算法,并对其进行改进。

### 1 基于阈值的图像分割算法

#### 1.1 最大熵法

Kapur 等在 1985 年提出最大熵阈值分割算法<sup>[3]</sup>,此算法将目标和背景视为两个独立源,求目标熵和背景熵的和,取其最大值作为图像的最佳阈值。目标熵和背景熵表示为

$$H_f(T) = - \sum_{g=0}^T \frac{P(g)}{P(T)} \log \frac{P(g)}{P(T)}$$
$$H_b(T) = - \sum_{g=T+1}^{l-1} \frac{P(g)}{1-P(T)} \log \frac{P(g)}{1-P(T)}$$
$$T_{opt} = \operatorname{argmax}[H_f(T) + H_b(T)]$$

#### 1.2 最小误差法

Kittler 和 Illingwo 在 1986 年提出最小误差法<sup>[4]</sup>,其分割的准则函数为

$$J(t) = 1 + 2[P_0(t)\ln\sigma_0(t) + P_1(t)\ln\sigma_1(t)] - 2[P_0(t)\ln P_0(t) + P_1(t)\ln P_1(t)]$$

其中,  $P_0(t) = \sum_{i=0}^t h(i)$ ,  $P_1(t) = \sum_{i=t+1}^{l-1} h(i)$ ,  $\mu_0(t) = \frac{\sum_{i=0}^t ih(i)}{P_0(t)}$ ,

$$\mu_1(t) = \frac{\sum_{i=t+1}^{l-1} ih(i)}{P_1(t)}$$
$$\sigma_0^2(t) = \frac{\sum_{i=0}^t [i - \mu_0(t)]^2 h(i)}{P_0(t)}$$

$$\sigma_1^2(t) = \frac{\sum_{i=t+1}^{l-1} [i - \mu_1(t)]^2 h(i)}{P_1(t)}$$

得到的最佳阈值为  $T_{opt} = \operatorname{argmin} J(t)$ 。

### 2 基于边缘检测的分割算法

边缘检测的实质是采用某种算法来提取出图像中对象与背景间的交接线。图像灰度的变化情况可以用图像灰度分布的梯度来描述,因此可以用局部图像微分来获得边缘检测算子。

#### 2.1 Sobel 算子

Roberts 算子<sup>[5]</sup>的一个主要问题是计算方向差时对噪声敏感。Sobel 提出一种将方向差运算与局部平均相结合的方法。

到稿日期:2008-11-15 本文受重庆市科委自然科学基金计划资助项目(No. CSTC 2007BB2451)资助。

瞿 中(1972-),男,博士研究生,主要研究方向为数字图像处理、普适计算、分布式及并行处理等,E-mail:quzhong@cqupt.edu.cn.

法,即 Sobel 算子。该算子是在以  $f(x,y)$  为中心的  $3 \times 3$  邻域上计算  $x$  和  $y$  方向的偏导数,即

$$S_x = \{f(x+1,y-1) + 2f(x+1,y) + f(x+1,y+1)\} - \{f(x-1,y-1) + 2f(x-1,y) + f(x-1,y+1)\}$$

$$S_y = \{f(x-1,y+1) + 2f(x,y+1) + f(x+1,y+1)\} - \{f(x-1,y-1) + 2f(x,y-1) + f(x+1,y-1)\}$$

其梯度大小为

$$g(x,y) = \sqrt{S_x^2 + S_y^2}, \text{ 或 } g(x,y) = |S_x| + |S_y|$$

它的卷积算子为

$$\begin{pmatrix} 1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{pmatrix}$$

由上面两个卷积算子对图像运算后,代入  $g(x,y) = |S_x| + |S_y|$ , 可求得图像的梯度赋值  $g(x,y)$ , 然后选取适当门限  $TH$ , 并进行判断。如果  $(i,j)$  为阶跃状边缘点,  $\{g(i,j)\}$  为一个二值图像, 也就是图像的边缘图像。

## 2.2 Laplace 算子

Laplace 算子<sup>[6]</sup>对不同的边缘类型有 2 种估算模板:

$$\begin{pmatrix} 1 & 0 & 1 \\ 0 & -4 & 0 \\ 1 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 1 & 1 \\ 1 & -8 & 1 \\ 1 & 1 & 1 \end{pmatrix} \begin{pmatrix} -1 & 0 & -1 \\ 0 & 4 & 0 \\ -1 & 0 & -1 \end{pmatrix} \begin{pmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{pmatrix}$$

(1) 阶跃边缘

(2) 屋顶边缘

对阶跃状边缘,二阶导数在边缘点产生一个陡峭的零交叉, Laplace 算子就是据此对  $f(x,y)$  的每个像素取它关于  $x$  方向和  $y$  方向的二阶差分之和,这是一个与边缘方向无关的边缘检测算子。而对屋顶状边缘,在边缘点的二阶导数取极小值,这时对  $f(x,y)$  每个像素取它关于  $x$  方向和  $y$  方向的二阶差分之和的相反数。

分析 Laplace 算子,其算法的核心在于对 Laplace 算子  $\nabla^2 f(x,y)$  的估算。在已有的算法中,都以被检测像素为中心,作出在其等  $45^\circ$  角 8 个方向上进行检测的模板,不同之处在于使用的参数不同。在改进的 Laplace 算法中,使用的新模板为

$$\frac{1}{8} \begin{pmatrix} 0 & 1 & 0 & 1 & 0 \\ 1 & 4 & 8 & 4 & 1 \\ 0 & 8 & -56 & 8 & 0 \\ 1 & 4 & 8 & 4 & 1 \\ 0 & 1 & 0 & 1 & 0 \end{pmatrix}$$

除了原来的方向外,又增加了 8 个方向,共有等  $22.5^\circ$  角 16 个方向上进行检测的模板,并根据 Laplace 算子的可靠性设定了适当的权向量。

## 2.3 Prewitt 算子

Prewitt 提出了类似的计算偏微分估计值的方法<sup>[7]</sup>,

$$P_x = \{f(x+1,y-1) + f(x+1,y) + f(x+1,y+1)\} - \{f(x-1,y-1) + f(x-1,y) + f(x-1,y+1)\}$$

$$P_y = \{f(x-1,y+1) + f(x,y+1) + f(x+1,y+1)\} - \{f(x-1,y-1) + f(x,y-1) + f(x+1,y-1)\}$$

其梯度大小为

$$g(x,y) = \sqrt{P_x^2 + P_y^2}, \text{ 或 } g(x,y) = |P_x| + |P_y|$$

它的卷积算子为

$$\begin{pmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ -1 & -1 & -1 \end{pmatrix}$$

当用两个掩模板组成边缘检测器时,通常取较大的幅度作为输出值。将 Prewitt 算子扩展到 8 个方向,这些算子模板由理想的边缘子图像构成,依次用边缘模板去检测图像,与被检测区域最为相似的模板给出最大值。用这个最大值作为算子的输出值  $P(i,j)$ , 这样可将边缘像素检测出来。定义 Prewitt 边缘检测算子模板为<sup>[7]</sup>

$$\begin{pmatrix} 1 & 1 & 1 \\ 1 & -2 & 1 \\ -1 & -1 & -1 \end{pmatrix} \begin{pmatrix} 1 & 1 & 1 \\ 1 & -2 & -1 \\ 1 & -1 & -1 \end{pmatrix} \begin{pmatrix} 1 & 1 & -1 \\ 1 & -2 & -1 \\ 1 & 1 & -1 \end{pmatrix} \begin{pmatrix} 1 & -1 & -1 \\ 1 & -2 & -1 \\ 1 & 1 & 1 \end{pmatrix}$$

(1) 1 方向 (2) 2 方向 (3) 3 方向 (4) 4 方向

$$\begin{pmatrix} -1 & -1 & -1 \\ 1 & -2 & 1 \\ 1 & 1 & 1 \end{pmatrix} \begin{pmatrix} -1 & -1 & 1 \\ -1 & -2 & 1 \\ 1 & 1 & 1 \end{pmatrix} \begin{pmatrix} -1 & 1 & 1 \\ -1 & -2 & 1 \\ -1 & 1 & 1 \end{pmatrix} \begin{pmatrix} 1 & 1 & 1 \\ -1 & -2 & 1 \\ -1 & -1 & 1 \end{pmatrix}$$

(5) 5 方向 (6) 6 方向 (7) 7 方向 (8) 8 方向

## 3 最大类间方差算法及其改进

1979 年, N. Otsu 提出最大类间方差法(称为 Otsu 法)<sup>[8]</sup>, 计算简单、稳定有效,一直被广为使用,是一种性能良好的自动阈值选择方法,时间消耗明显少于其它阈值化算法。

设待分割图像的像素数为  $N$ , 它有  $L$  个灰度级  $(0, 1, \dots, L-1)$ , 灰度级为  $i$  的像素数为  $n_i$ , 那么  $N = \sum_{i=0}^{L-1} n_i$ , 直方图表示为概率密度分布  $p_i = \frac{n_i}{N}$ ,  $\sum_{i=0}^{L-1} p_i = 1$ ,  $p_i \geq 0$ 。

设阈值  $t$  将图像分成两类  $C_0$  和  $C_1$  (目标和背景), 则  $C_0$  和  $C_1$  分别对应具有灰度级  $\{0, 1, \dots, t\}$  和  $\{t+1, t+2, \dots, L-1\}$  的像素。设  $\sigma_B^2(t)$  表示直方图中阈值为  $t$  时的类间方差, 则最佳阈值可以通过求  $\sigma_B^2(t)$  的最大值而得到, 即

$$\sigma_B^2(t^*) = \max_{0 \leq t \leq L-1} \{\sigma_B^2(t)\}$$

其中,

$$\sigma_B^2(t) = \omega_0(t) [\mu_0(t) - \mu_T]^2 + \omega_1(t) [\mu_1(t) - \mu_T]^2,$$

$$\sigma_T = \sum_{i=1}^K (i - \mu_T)^2 P_i, \eta = \max \frac{\sigma_B}{\sigma_T},$$

$$\mu_T = \sum_{i=0}^{L-1} i P_i, \omega_0(t) = \sum_{i=0}^t P_i, \omega_1(t) = 1 - \omega_0(t),$$

$$\mu(t) = \sum_{i=0}^t i P_i, \mu_0(t) = \frac{\mu(t)}{\omega_0(t)}, \mu_1(t) = \frac{\mu_T - \mu(t)}{1 - \omega_0(t)}.$$

Otsu 算法的不足:

1) 传统 Otsu 算法在处理灰度级不连续的图像时, 所求出的阈值不能很好地收敛到全局最优。

2) 尽管 Otsu 算法没有对概率密度函数做任何假设, 仅仅利用均值和方差表达了背景和背景这两个概率密度函数, 因此它假设了利用这两个统计量足以表示这两个概率密度函数, 但事实上并非如此。

3) 当背景和目标的两个总体分布差异很大时, Otsu 算法将会失效。

4) Otsu 算法适用于图像中有两个类别; 当图像类多于两个时, 这种方法必须修改, 以确定多个阈值。这种方法使得类间方差最大, 而类内方差最小。

5) Otsu 算法将图像分为了两类, 即使这种分法没有任何实际意义。事实上, 当一幅图像在变化的光照条件下该方法不能直接应用。

改进的 Otsu 算法:

① 计算图像灰度级  $L$ 。计算图像的灰度平均值  $\mu_T$ , 将  $\mu_T$

取整 $\lfloor \mu_T \rfloor$ 作为图像的灰度级,即 $L = \lfloor \mu_T \rfloor$ ;

②分割次数赋初值。 $J = 1$ ;

③Otsu算法实现。利用Otsu算法计算图像的像素 $N$ 、阈值 $K$ 、阈值选择函数 $\eta$ 、类内方差 $\sigma_w$ ;

④迭代。

$N(J) = N; K(J) = K; L = K$ ;

$\eta(J) = \eta; \sigma(J) = \sigma_w; J = J + 1$ ;

⑤if  $J <= 1$  then goto ③;

⑥计算 $\epsilon$ 。

$$\epsilon = \frac{[N(J-1) - N(J)]}{N(J-1)} - \frac{[\sigma(J-1) - \sigma(J)]}{\sigma(J-1)}$$

⑦if  $\epsilon >= 0$  then goto ③;

⑧查找最大 $\eta(J)$ 所对应的阈值,作为最佳阈值 $K$ 。

#### 4 仿真实验及其结果分析

为了对比各种算法,本文采用图1所示的Lena.tif和Mri.jpg作为测试图像,运用不同的方法进行仿真实验。



图1 Lena.tif和Mri.jpg测试图像

图1 Lena.tif和Mri.jpg测试图像

首先通过实验可以得到图2所示的两幅图像的直方图。

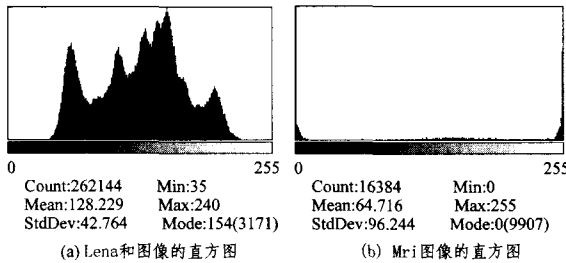


图2 Lena.tif和Mri.jpg的直方图

##### 4.1 仿真实验

运用最大熵法得到的仿真图像如图3所示。



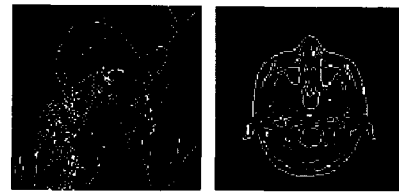
图3 最大熵法得到的图像

运用最小误差法仿真结果如图4所示。

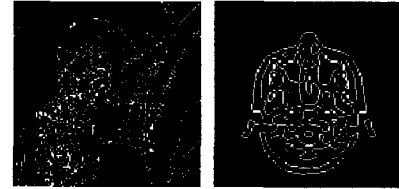


图4 最小误差法得到的图像

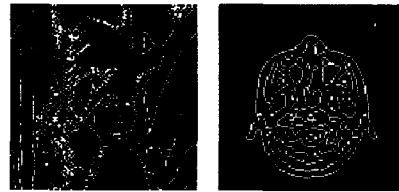
运用Sobel算子、改进的Laplace算子和Prewitt算子进行边缘检测,得到的图像边缘如图5所示。



(a) Sobel算子



(b) Laplace算子



(c) Prewitt算子

图5 不同边缘检测得到的图像

运用改进的Otsu法得到的仿真图像如图6所示。



图6 最大类间方差法得到的图像

##### 4.2 实验结果分析

采用改进的最大类间方差、最大熵和最小误差阈值分割方法,针对Lena和Mri图像,得到的阈值如表1所列。

表1 不同的方法得到的阈值

	改进的最大类间方差算法	最大熵算法	最小误差算法
Lena.tif	117	123	91
Mri.jpg	101	136	1

从图3至图6、表1中可以看出,分别采用了改进的最大类间方差、最大熵、最小误差阈值分割方法和边缘检测算法,得到了不同的分割结果。

1) 针对Lena.tif图像,Otsu法的分割结果最好,不但完整地提取出了整个目标,而且误分入目标的背景也比较少。

2) 对Mri.jpg图像,最大熵法分割结果最好,但图像的中间部分存在模糊现象,分割效果还有待提高。

3) 从图5可以看出,Prewitt算子对图像的靠上和靠下部分的检测比较好,而对中间部分的图像检测效果不明显,存在模糊现象;Sobel算子和Laplace算子的边缘检测效果不明显。

4) 在改进的Laplace算子中,可以提高边缘检测的精度,同时由于合理地设置了参数,因而避免了一些伪边缘的提取。改进的Laplace算子相对于原来的Laplace算子而言,

(下转第290页)

为了测试流并行加速的效果,我们分别统计了单个计算核心所获得的加速比以及整个应用的加速比,主要是将 Isim 模拟器上运行程序的绝对时间与通用处理器上执行串行程序的绝对时间进行比较。单个计算核心的加速性能主要是由大量的运算部件以及 VLIW 技术所带来的,取决于核心内部的数据并行性以及指令并行性。整个应用的加速性能则从全局进行系统考虑,包括各种各样的组织调度开销,有些应用虽然单个核心的加速效果不错,但由于在组织调度上花费了太多的时间,最终的应用加速比也可能不会太好。图 7 为我们得到的统计结果。

应用程序名	主要计算核心	核心加速比	应用加速比
JPEG	color	30.5	8.6
	DCT	72.6	
	dc_ac_code	13.4	
MPEG	blocksearch_kc	23.5	5.3
	diff_kc	50.2	
	dct_kc	9.8	
	rle_kc	10.6	
Rasta	lbrand_kc	10.6	2.1
	ldft_kc	20.8	
	filt_kc	11.3	
	Powspec_kc	11.3	

图 7 多媒体应用的流并行加速性能

从图 7 中可以看出, JPEG 程序单个计算核心的加速性能较好,最终的应用加速比也相对较高,说明 JPEG 具有比较自然的数据分块特性,数据块上的计算比较密集,对数据的访存也比较规则,所以用于额外的组织调度的开销比较少,得到了比较理想的加速比; MPEG2 中由于相邻图像帧之间的依赖关系形成的反馈环在一定程度上破坏了应用的生产者-消费者局部性,而且有相当一部分计算发生在标量机上,所以虽然单个计算核心的加速比较好,整个应用加速比却只达到了 5 左右; RASTA 的性能最差,短流、计算不密集的核心使得核

心加速比相对较低,而不规则的访存和控制又造成了大量的组织调度开销,所以其应用加速比也不高。

**结束语** 通过上述的理论分析以及实验测评,可以看出流体系结构对媒体处理做了很好的软硬件支持。利用流编程模型可以很自然地开发多媒体应用中存在的流并行性。但是一个具体的应用能否在流体系结构上取得好的流并行效果,又取决于多个方面的因素,包括多媒体应用本身具有的特点、程序设计的方法以及使用的并行算法等等。

## 参考文献

- [1] Kapasi U J, Dally W J. The Imagine Stream Processor[C]// Proceedings of the 2002 International Conference on Computer Design. September 2002: 16-18
- [2] Lee Chunho, Potkonjak M, Mangione-Smith W H. MediaBench: A Tool for Evaluating and Synthesizing Multimedia and Communications Systems[C]// Proceedings of the International Symposium on Microarchitecture. IEEE Micro-30. IEEE Computer Society Press, 1997: 330-335
- [3] Fritts J, Wolf W, Liu B. Understanding multimedia application characteristics for designing programmable media processors[C]// SPIE Media Processors '99. January 1999
- [4] William S A. Stream Languages and Programming Models[C]// PACT 2003. September 2003
- [5] 郝杰, 吴元清. 实用多媒体技术及其 C 语言实现[M]. 北京: 电子工业出版社, 1995
- [6] 精英科技, 等. 视频压缩与音频编码技术[M]. 北京: 中国电力出版社, 2001
- [7] Lawrence Rabiner, Bing-Hwang Juang. 语音识别基本原理[M]. 北京: 清华大学出版社, 1999

(上接第 278 页)

不但检测出来的边缘更清晰,而且检测出原来没有检测出的一些边缘。

**结束语** 阈值化技术和特征空间聚类都属于并行区域分割技术,是图像分割中最重要的而且有效的技术之一,在实际的图像处理系统中得到了广泛应用。特别是在需要实时性较强的图像处理系统中,快速而准确的图像阈值化方法就成为非常重要的研究目标。对一幅具体的图像,选用何种算法,要进行对比实验,不存在一种通用的图像分割算法。

## 参考文献

- [1] 秦安,冯前进,陈武凡. MR 心脏序列图像左心室内外壁联合分割和时序追踪新方法[J]. 中国图象图形学报, 2008, 13(1): 80-88
- [2] 贾丽娟,张光年,葛庆平,等. 一种改进的彩色图像分割算法[J].

计算机工程与应用, 2008, 44(13): 159-160, 163

- [3] Kapur J N, Sahoo P K, Wong A K C. A New Method for Gray-level Picture Thresholding Using the Entropy of the Histogram [J]. Graphical Models and Image Processing, 1985, 29: 273-285
- [4] Kittler J, Illingworth J. Minimum Error Thresholding [J]. Pattern Recognition, 1986, 19: 41-47
- [5] 黄伟,周鸣争,李小牛. 一种基于四元数的彩色图像边缘检测改进算法 [J]. 计算机研究与发展, 2008, 45(3): 121-124
- [6] 徐永生. 基于小波分析的大气颗粒物数字图像的边缘检测[D]. 武汉: 武汉理工大学, 2006
- [7] 赵海燕. 利用改进的 Prewitt 边缘算子进行车牌定位[J]. 长春理工大学学报, 2005, 28(1): 50-51, 46
- [8] Otsu N. A threshold selection method from gray-level histogram [J]. IEEE Trans. Systems Man Cybernet, 1979, 9(1): 62-66