

基于边缘检测与 Otsu 的图像分割算法研究

杨 陶, 田怀文, 刘晓敏, 柯小甜, 高松松, 马梦婕

(西南交通大学 机械工程学院, 成都 610031)

摘 要: Otsu 算法作为图像分割领域的经典算法得到了广泛的应用, 在其基础上发展起来的 2 维 Otsu 算法由于运算时间长、抗噪能力差, 应用受到限制。为此, 提出一种改进的 Otsu 算法。通过更改 2 维直方图的区域划分方式, 分别运用 Sobel, Log 和 Canny 边缘检测算法与直线拟合法相结合, 将图像的目标和背景区域限制在一对平行于对角线的界线内, 使用噪声点的邻域均值代替其灰度值, 利用 2 维 Otsu 斜分法将目标从背景中分割出来。实验结果表明, 与传统 2 维 Otsu 算法及其改进算法相比, 该算法不仅运算时间较短, 而且具有较好的分割质量、抗噪性能和自适应能力。

关键词: 图像分割; 边缘检测; 直线拟合; 直方图; Otsu 算法

中文引用格式: 杨 陶, 田怀文, 刘晓敏, 等. 基于边缘检测与 Otsu 的图像分割算法研究[J]. 计算机工程, 2016, 42(11): 255-260, 266.

英文引用格式: Yang Tao, Tian Huaiwen, Liu Xiaomin, et al. Research on Image Segmentation Algorithm Based on Edge Detection and Otsu[J]. Computer Engineering, 2016, 42(11): 255-260, 266.

Research on Image Segmentation Algorithm Based on Edge Detection and Otsu

YANG Tao, TIAN Huaiwen, LIU Xiaomin, KE Xiaotian, GAO Songsong, MA Mengjie

(School of Mechanical Engineering, Southwest Jiaotong University, Chengdu 610031, China)

[Abstract] As a classical algorithm, Otsu algorithm is applied widely in image segmentation field. The application of Two-dimensional(2D) Otsu algorithm based on Otsu algorithm is limited for the long computing time and poor anti-noise capacity. Therefore, this paper proposes a modified Otsu algorithm. By using new area partition method, the proposed algorithm respectively applies three edge detection operators (Sobel, Log and Canny) combined with linear fitting method to limit the object and background between a pair of boundaries which are parallel to the diagonal, and then uses the domain average value of noise pixels instead of the grey value. Finally the 2D Otsu oblique segmentation method is used to separate object from background. Experimental results show that compared with the traditional 2D Otsu algorithm and its modified algorithms, the proposed algorithm not only has a short operation time but also has a high quality of segmentation, a strong anti-noise capacity and a preferable adaptive ability.

[Key words] image segmentation; edge detection; linear fitting; histogram; Otsu algorithm

DOI:10.3969/j.issn.1000-3428.2016.11.042

0 概述

图像分割是图像处理与机器视觉的重要组成部分, 随着工业生产智能化的不断推进, 历年来一直受到众多国内外学者的高度关注。从 20 世纪 60 年代以来, 国内外学者提出了许多图像分割算法, 主要包

括基于阈值、基于区域、基于边缘和基于特定理论的分割方法^[1], 其中, Otsu 阈值分割法因其简单、高效的特性而成为应用最广泛的算法之一。

Otsu 算法是由日本人大津在 1979 年提出的, 也称大津法或最大类间方差法^[2]。它在处理无噪声干扰且直方图具有明显双峰的图像时效果很理想。但

基金项目: 国家自然科学基金(51275431)。

作者简介: 杨 陶(1990—), 男, 硕士研究生, 主研方向为图像分割、机器视觉; 田怀文, 教授、博士; 刘晓敏、柯小甜、高松松、马梦婕, 硕士。

收稿日期: 2015-10-12 **修回日期:** 2015-12-15 **E-mail:** ytswjtu@sina.com

由于图像来源千差万别,图像在拍摄时易受噪声干扰,导致直方图分布形式多样化,采用简单的 Otsu 算法无法得到满意的结果。为了提高分割结果的准确性,文献[3]引入了邻域均值,在 1 维 Otsu 算法的基础上提出了 2 维 Otsu 算法。

然而,传统的 2 维 Otsu 算法将对角线上两区域的概率和视为 1,但实验证明该假设不合理^[4],算法的计算量也较大。为了克服这些缺点,文献[5]提出了基于遗传算法的 2 维 Otsu 算法,文献[6]提出了 2 维自适应阈值的快速算法,2 种方法在降低计算量的同时也改善了分割质量,但抗噪能力依然不足,通过人为确定目标和背景的带状区域大小,一般需要经过多次调整、反复试验,适应性较差。为此,本文提出一种将边缘检测^[7]与 2 维 Otsu 斜分法^[8]相结合的图像分割法。

1 2 维 Otsu 阈值分割理论

设一幅分辨率为 $M \times N$ 的图像灰度级为 L ,则其邻域均值图像(以 3×3 的邻域模板获取邻域均值)灰度级也为 L 。图像中的每个像素对应一个二元组 (i, j) , $i = 0, 1, \dots, L-1$; $j = 0, 1, \dots, L-1$, 其中, i 为像素灰度值; j 为邻域均值。设像素灰度值为 i 且邻域均值为 j 的像素点个数为 r_{ij} , 则 2 维联合概率密度为:

$$P_{ij} = \frac{r_{ij}}{M \times N} \quad (1)$$

任意给定一个阈值点 (s, t) 就可在 2 维直方图中将图像分为 4 个区域,如图 1 所示。其中, D_0, D_1 分别是目标和背景区域; A, B 内包含边缘和噪声。

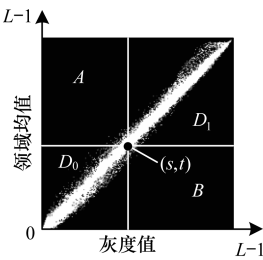


图 1 2 维直分直方图

整个 2 维直方图的概率和为:

$$\sum_{i=0}^{L-1} \sum_{j=0}^{L-1} p_{ij} = 1 \quad (2)$$

区域 D_0, D_1 的概率分别被记为 ω_0, ω_1 , 则:

$$\omega_0 = \sum_{i=0}^{s-1} \sum_{j=0}^{t-1} p_{ij} \quad (3)$$

$$\omega_1 = \sum_{i=s}^{L-1} \sum_{j=t}^{L-1} p_{ij} \quad (4)$$

对应的灰度级均值矢量为:

$$\mu_0 = (\mu_{0i}, \mu_{0j}) = \left(\sum_{i=0}^{s-1} \sum_{j=0}^{t-1} \frac{ip_{ij}}{\omega_0}, \sum_{i=0}^{s-1} \sum_{j=0}^{t-1} \frac{jp_{ij}}{\omega_0} \right) \quad (5)$$

$$\mu_1 = (\mu_{1i}, \mu_{1j}) = \left(\sum_{i=s}^{L-1} \sum_{j=t}^{L-1} \frac{ip_{ij}}{\omega_1}, \sum_{i=s}^{L-1} \sum_{j=t}^{L-1} \frac{jp_{ij}}{\omega_1} \right) \quad (6)$$

那么,图像的总均值矢量为:

$$\mu_z = (\mu_{zi}, \mu_{zj}) = \left(\sum_{i=0}^{L-1} \sum_{j=0}^{L-1} ip_{ij}, \sum_{i=0}^{L-1} \sum_{j=0}^{L-1} jp_{ij} \right) \quad (7)$$

传统的 2 维 Otsu 法忽略了非对角线区域 A, B 的概率,假设区域 D_0, D_1 的概率和为 1,即:

$$\omega_0 + \omega_1 \approx 1 \quad (8)$$

则类间方差可表示为:

$$d(s, t) = \omega_0 [(\mu_{0i} - \mu_{zi})^2 + (\mu_{0j} - \mu_{zj})^2] + \omega_1 [(\mu_{1i} - \mu_{zi})^2 + (\mu_{1j} - \mu_{zj})^2] \quad (9)$$

因此,最佳阈值的选取原则为:

$$(s^*, t^*) = \arg \max_{0 \leq s, t \leq L-1} \{d(s, t)\} \quad (10)$$

2 本文算法

2.1 改进的 2 维斜分直方图

从图 1 可以看出,概率不为 0 的像素点都集中在对角线附近的一条带状区域内。2 维 Otsu 算法假设 A, B 的概率和为 0,但该假设在对角线附近并不成立,这样会导致阈值分割结果不准确。此外,2 维 Otsu 算法的阈值搜索过程需要经过 L^2 次循环,计算量很大,不适用于实时处理图像。因此,有必要对 2 维直方图进行重新分割。

根据像素灰度分布规律,文献[6]将 2 维直方图按图 2(a)的方式分割成 4 个区域。其中, D_0, D_1 分别是目标和背景区域; A, B 是噪声区域。2 条界线 $g - f \pm m = 0$ 限定了 D_0, D_1 的带宽,垂直于对角线的直线 $g + f - T = 0$ 为分割直线, T 为阈值。该分区方式假设 2 条界线关于对角线对称。实际上这种假设不太合理,这是因为图像的像素点不是关于 2 维直方图的对角线对称分布。

本文采用了一种界线非对称的直方图斜分方式,如图 2(b)所示。只有当 $m = n$ 时,2 条界线才关于对角线对称。由于图像边缘点的灰度值会在一定程度上高于或低于其邻域内其余像素的灰度值,因此其灰度值和邻域均值也存在一定偏差。因此,在 2 维直方图中,边缘点会适当偏离对角线,并介于噪声区域与目标背景区域之间^[8]。当边缘点的灰度值大于其邻域均值时,边缘点位于对角线下方;反之,边缘点位于对角线上方。本文通过分别拟合对角线上下两侧的边缘点而得到 2 条直线,即 A, B 区域与 D_0, D_1 区域之间的界线。

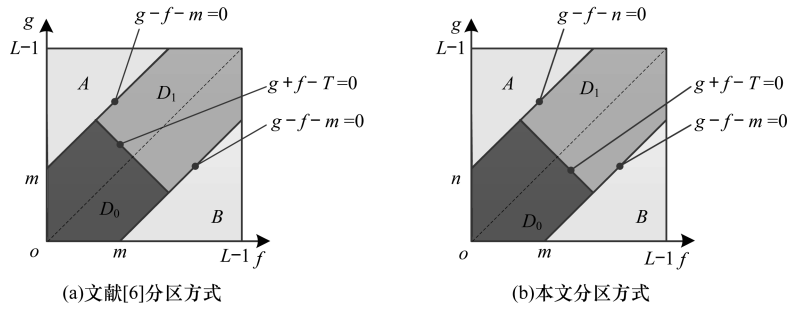


图 2 2 维斜分直方图

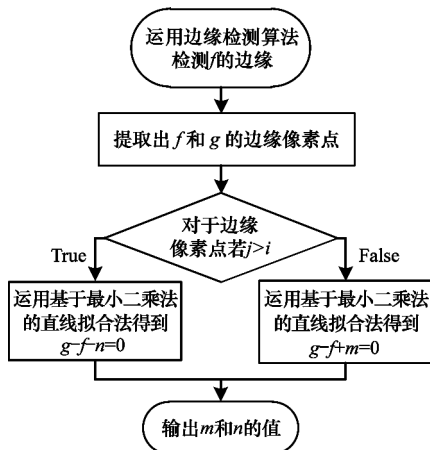
为了提高该算法的抗噪能力,对于 A, B 中的噪声点 (x, y) , 用其邻域均值代替灰度值, 即当 $g - f - n > 0$ 或 $g - f + m < 0$ 时, 令:

$$f(x, y) = g(x, y) \quad (11)$$

将噪声替换后, 在 2 维直方图中, 所有噪声点将从 A, B 转移到对角线上, 成为目标和背景区域的一部分。然后运用 Otsu 斜分法进行阈值分割, 可以提高算法的抗噪性能。

2.2 带状区域大小确定

图 3 是计算 m, n 值的算法流程, 即确定带状区域大小的过程。本文采用对比分析的方式分别运用 Sobel, Log 和 Canny 边缘检测算法获取目标和背景区域的界线。

图 3 获得 m, n 值的算法流程

首先分别用这 3 种算法检测原始图像 f 的边缘, 获取所有边缘点。对于每个边缘点, 其灰度值为 i , 邻域均值为 j 。那么在 2 维直方图中对应坐标点 (i, j) , 边缘点会分散在对角线两侧。对于对角线左上侧的边缘点, 运用基于最小二乘法的直线拟合法得到直线 $g - f - n = 0$ 。对于对角线右下侧的边缘点, 同理得到直线 $g - f + m = 0$, 即得 m, n 的值。用该方式确定带状区域大小不仅能增强算法的自适应能力, 而且能提高分割质量。

2.3 2 维 Otsu 斜分法的快速实现

假设 $\omega_0(T), \mu_{0i}(T), \mu_{0j}(T)$ 分别代表 D_0 类的概率和 2 个均值, $\omega_1(T), \mu_{1i}(T), \mu_{1j}(T)$ 分别代表 D_1 类的概率和 2 个均值, μ_i, μ_j 代表 2 个总均值。

当 $0 \leq T \leq L-1$ 时, 如图 2(b) 所示。有:

$$\begin{aligned} \omega_0(T) &= \sum_{i=0}^T \sum_{j=0}^{T-i} P(i, j) \\ &= \sum_{i=0}^{T-1} \sum_{j=0}^{T-1-i} P(i, j) + \sum_{i=0}^T P(i, T-i) \\ &= \omega_0(T-1) + \sum_{i=0}^T P(i, T-i) \end{aligned} \quad (12)$$

$$\begin{aligned} \mu_{0i}(T) &= \sum_{i=0}^T \sum_{j=0}^{T-i} iP(i, j) \\ &= \sum_{i=0}^{T-1} \sum_{j=0}^{T-1-i} iP(i, j) + \sum_{i=0}^T iP(i, T-i) \\ &= \mu_{0i}(T-1) + \sum_{i=0}^T iP(i, T-i) \end{aligned} \quad (13)$$

$$\begin{aligned} \mu_{0j}(T) &= \sum_{i=0}^T \sum_{j=0}^{T-i} jP(i, j) \\ &= \sum_{i=0}^{T-1} \sum_{j=0}^{T-1-i} jP(i, j) + \sum_{i=0}^T (T-i)P(i, T-i) \\ &= \mu_{0j}(T-1) + \sum_{i=0}^T (T-i)P(i, T-i) \end{aligned} \quad (14)$$

同理可得: $L-1 < T < 2L-2$ 时的递推公式。

则有:

$$\omega_0(T) = \begin{cases} \omega_0(T-1) + \sum_{i=0}^T P(i, T-i), & 0 \leq T \leq L-1 \\ 1 - \omega_1(T), & L-1 < T < 2L-2 \end{cases} \quad (15)$$

其中, $\omega_1(T) = \omega_1(T+1) + \sum_{i=T-L+1}^{L-1} P(i, T-i)$ 。

$$\mu_{0i}(T) = \begin{cases} \mu_{0i}(T-1) + \sum_{i=0}^T iP(i, T-i), & 0 \leq T \leq L-1 \\ \mu_i - \mu_{1i}(T), & L-1 < T < 2L-2 \end{cases} \quad (16)$$

其中, $\mu_{1i}(T) = \mu_{1i}(T+1) + \sum_{i=T-L+1}^{L-1} iP(i, T-i)$, $\mu_i = \sum_{i=0}^{L-1} \sum_{j=0}^{L-1} iP(i, j)$ 。

$$\mu_{0j}(T) = \begin{cases} \mu_{0j}(T-1) + \sum_{i=0}^T (T-i)P(i, T-i), & 0 \leq T \leq L-1 \\ \mu_j - \mu_{1j}(T), & L-1 < T < 2L-2 \end{cases} \quad (17)$$

其中, $\mu_{ij}(T) = \mu_{ij}(T+1) + \sum_{i=T-L+1}^{L-1} (T-i)P(i, T-i), \mu_j$
 $= \sum_{i=0}^{L-1} \sum_{j=0}^{L-1} jP(i, j)。$

特别地, 有:

$$\omega_0(0) = P(0, 0), \omega_1(2L-2) = P(L-1, L-1)$$

$$\mu_{0i}(0) = 0, \mu_{1i}(2L-2) = (L-1)P(L-1, L-1)$$

$$\mu_{0j}(0) = 0, \mu_{1j}(2L-2) = (L-1)P(L-1, L-1)$$

类间方差表示为:

$$t_r S_B(T) = \frac{[(\mu_i \omega_0(T) - \mu_{0i}(T))^2 + (\mu_j \omega_0(T) - \mu_{0j}(T))^2]}{\omega_0(T)(1 - \omega_0(T))} \quad (18)$$

当类间方差最大时取得最佳阈值 T^* , 即:

$$T^* = \arg \max_{0 < T < 2L-2} \{t_r S_B(T)\} \quad (19)$$

3 实验结果与分析

本文实验设备为 Intel Core™ i5-4210U CPU @ 1.70 GHz, 4 GB 内存的笔记本电脑, 软件采用 Matlab R2010b。测试实验分为 2 组, 即无噪图像分割实验测试算法的分割质量和分割速度, 有噪图像分割实验测

试算法的抗噪性能。为了让本文算法更具说服力, 将 3 种算法 (Sobel + Otsu, Log + Otsu 和 Canny + Otsu) 的分割结果与文献[8-11]所改进的 Otsu 算法以及传统的 2 维 Otsu 快速算法^[12]进行对比。

3.1 无噪图像分割实验

本文对多张无噪图像进行了测试, 图 4 是其中 4 张比较具有代表性的测试图像(图像 1~图像 4), 分辨率从左至右依次为 256 像素 × 256 像素, 364 像素 × 366 像素, 321 像素 × 481 像素, 512 像素 × 512 像素。各算法的分割结果和实验数据如图 5 和表 1 所示。从图 5 可以看出, 文献[11]与传统的 2 维 Otsu 快速算法分割结果相近且均不太理想。因为文献[11]与传统 2 维 Otsu 快速算法都是基于 2 维直分直方图的算法, 2 维直分直方图忽略了非对角线区域内的像素点。



图 4 4 张无噪图像



(a) 2 维 Otsu 快速算法



(b) 文献[8]算法



(c) 文献[9]算法



(d) 文献[10]算法



(e) 文献[11]算法



(f) Sobel+Otsu 算法



(g) Log+Otsu 算法



(h) Canny+Otsu 算法

图 5 各算法对无噪图像的分割结果

表 1 各算法对无噪图像分割的数据对比

算法	图像 1			图像 2			图像 3			图像 4		
	耗时/s	阈值	NU	耗时/s	阈值	NU	耗时/s	阈值	NU	耗时/s	阈值	NU
2 维 Otsu 快速算法	4.400	66,141	0.438	4.928	169,107	0.415	4.955	94,63	0.522	5.072	92,150	0.438
文献[8]算法	0.880	178	0.489	1.518	280	0.591	2.078	130	0.545	3.962	235	0.526
文献[9]算法	0.532	86	0.485	0.837	137	0.583	1.281	61	0.539	2.001	116	0.518
文献[10]算法	0.021	177	0.489	0.034	279	0.591	0.046	129	0.545	0.094	234	0.526
文献[11]算法	1.242	44,134	0.321	1.843	168,105	0.401	2.470	92,38	0.438	3.935	88,147	0.413
Sobel + Otsu 算法	0.097	177	0.495	0.233	277	0.627	0.267	130	0.618	0.565	234	0.556
Log + Otsu 算法	0.126	177	0.503	0.342	278	0.636	0.668	130	0.639	1.041	235	0.557
Canny + Otsu 算法	0.141	177	0.500	0.483	277	0.639	0.878	130	0.644	1.789	235	0.558

文献[9]的分割结果明显优于传统2维Otsu快速算法和文献[11]算法。因为文献[9]算法是基于1维直方图的二次分割,一定程度上提高了分割结果的准确性。本文的3种算法、文献[8,10]算法的分割结果较好且非常接近。因为这些算法都是基于2维斜分直方图的算法,该分区方式相对于直分分区更准确。但仔细观察会发现,本文3种算法对细节的分割效果更好。

为了更加直观地比较各算法的分割质量,本文引入区域内部均匀性(NU)^[13-14]作为量化指标。区域内部均匀性常常用来描述图像分割所形成的区域内部相似程度,其值越大说明区域内部的相似程度越高,从而说明分割质量越好。从表1可以看出,本文3种算法的区域内部均匀性很接近,且均在不同程度上高于其余算法,与视觉效果一致。

对算法的研究发现,算法耗时主要由直方图构建时间和阈值分割时间组成。从表1可以看出,文献[10]算法耗时最短。文献[10]从2维斜分直方图中派生出了1维直线截距直方图,在1维直方图的基础上进行阈值分割,因此,大大缩减了直方图构建时间和阈值分割时间。本文3种算法的耗时稍逊于文献[10]算法但优于其余算法。本文算法在计算阈值时,对于每个阈值 T ,只需用前面得到的 $\omega_0(T)$, $\mu_{0i}(T)$, $\mu_{0j}(T)$ 和 $\omega_1(T)$, $\mu_{1i}(T)$, $\mu_{1j}(T)$ 叠加直线段 $g+f-T=0$ 上各个点相应的概率值即可,且阈值搜索范围为 $(0, 2L-2)$,因此,本文算法的计算复杂度数量级仅为 $O(L)$ 。此外,由于替换了噪声点,噪声区域的概率为0,在构建2维直方图时只需计算带状区域内点的概率,有效地缩减了直方图的构建时间,这也是文献[8]算法速度低于本文算法的主要原因。文献[9]通过2次构建1维直方图并进行阈值分割,运算时间相对较短。文献[11]算法和传统的2维

Otsu快速算法均需构建2维直方图,且在进行阈值搜索时均需遍历2维直方图上的所有点,因此,算法耗时相对较长。

综合上看,本文的基于Sobel与Otsu的图像分割算法分割质量较高,算法耗时较短,在所有算法中表现最好。

3.2 有噪图像分割实验

由于实际拍摄的图像可能受到多种噪声干扰,且噪声强度较小,一般不会超过0.1。因此,该实验对多张带有强度为0.06的混合噪声(高斯噪声、椒盐噪声和乘性噪声的等强度混合)图像进行了测试,图6是其中,4张比较具有代表性的测试图像,分辨率从左至右依次为256像素×256像素,512像素×512像素,321像素×481像素,321像素×481像素。由于本文算法中提出将噪声点的灰度值用其邻域均值替换的降噪方式,该方式的思想类似于均值滤波,因此为了使本文算法的抗噪性能测试更具说服力,在运用其余算法之前,先对图像进行均值滤波。图7和表2是各算法对噪声图像的分割结果和实验数据。从图7可以看出,文献[11]和传统的2维Otsu快速算法的分割结果中噪声点较多,抗噪性能较差。文献[9]的分割结果中噪声点也较多,但稍少于文献[11]和传统的2维Otsu快速算法。文献[8,10]的分割结果中噪声点相对较少,具有一定的抗噪性能。相对来看,本文算法的分割结果中噪声点最少,说明降噪效果较好。

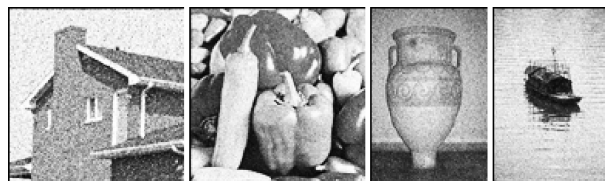


图6 噪声图像5~图像8

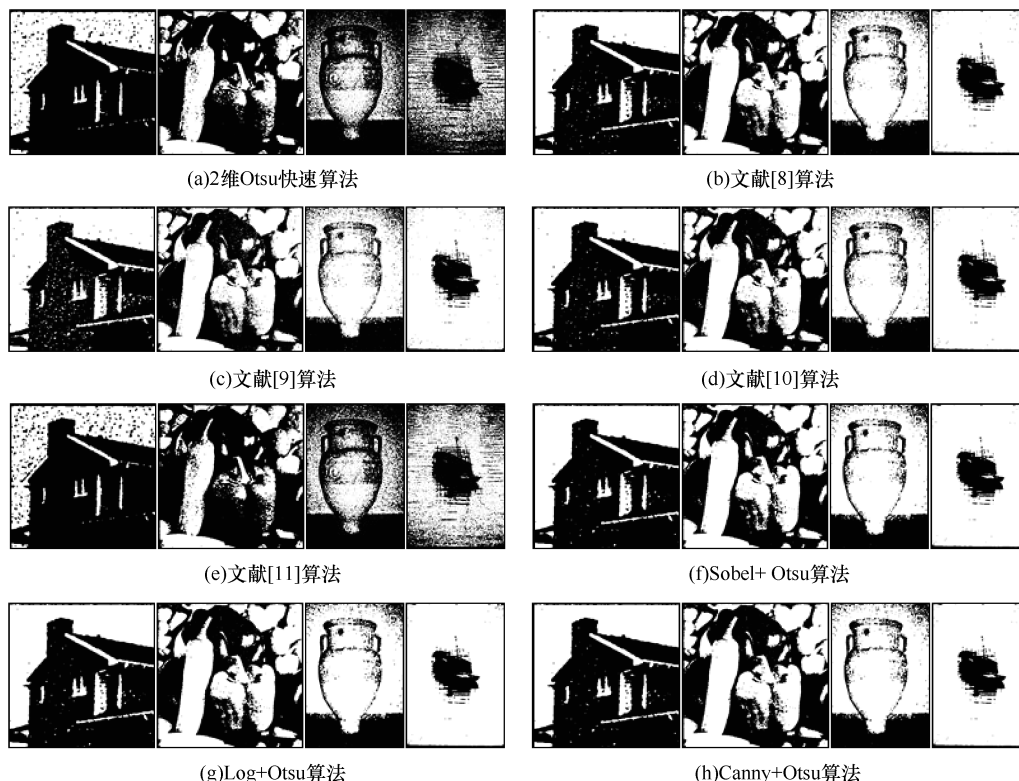


图 7 各算法对噪声图像的分割结果

表 2 各算法对椒盐噪声图像分割的数据对比

算法	图像 5		图像 6		图像 7		图像 8	
	阈值	PSNR/dB	阈值	PSNR/dB	阈值	PSNR/dB	阈值	PSNR/dB
2 维 Otsu 快速算法	120,173	14.471	92,152	12.571	132,93	7.998	163,101	6.940
文献[8]算法	300	16.710	250	14.284	213	11.409	220	18.829
文献[9]算法	147	14.913	122	12.944	105	10.741	108	16.653
文献[10]算法	299	16.605	249	14.260	212	11.541	219	18.964
文献[11]算法	123,177	13.318	95,155	12.460	128,85	8.420	146,74	8.528
Sobel + Otsu 算法	299	17.669	249	14.861	211	12.302	219	21.077
Log + Otsu 算法	299	17.688	249	14.914	211	12.363	220	21.249
Canny + Otsu 算法	299	17.696	249	14.962	211	12.369	219	21.216

为了更加直观地比较各算法的抗噪性能,本文引入峰值信噪比(Peak Signal to Noise Ratio, PSNR)^[15]作为量化指标。PSNR 值越大代表算法的抗噪性能越好。从表 2 可以看出,本文 3 种算法的 PSNR 值非常接近且均高于其余各算法。此外,其余各算法的 PSNR 值也与图 7 中的分割结果图保持一致。

4 结束语

本文针对 2 维直方图进行重新分区,采用边缘检测与直线拟合相结合的方式来确定包含目标和背景的带状区域大小,再用噪声点的邻域均值代替其灰度值,提出一种基于边缘检测与 Otsu 的图像分割算法。实验结果表明,该算法的运算时间短、自适应能力强,满足了图像处理的实时性要求。此外,在本文提出的 3 种算法中,基于 Sobel 与 Otsu 的图像分割算法计算复杂度最低,表现得尤为出色,具有一定

的推广价值。但本文算法对光照不均的图像分割效果不佳,因此,提高该类图像的分割质量是下一步研究的内容。

参考文献

- [1] Mehmet S, Bulent S. Survey over Image Thresholding Techniques and Quantitative Performance Evaluation[J]. Journal of Electronic Imaging, 2004, 13(1): 146-165.
- [2] Otsu N. A Threshold Selection Method from Gray-level Histograms[J]. IEEE Transactions on System Man and Cybernetic, 1979, 9(1): 62-66.
- [3] 刘健庄, 栗文青. 灰度图像的二维 Otsu 自动阈值分割法[J]. 自动化学报, 1993, 28(3): 1168-1169.
- [4] Chen Qi, Zhao Long, Lu Jun, et al. Modified Two-dimensional Otsu Image Segmentation Algorithm and Fast Realization[J]. IET Image Process, 2012, 6(4): 426-433.

(下转第 266 页)

6 结束语

本文提出一种新的旋转对称目标检测算法,使用旋转对称隐模型表示方法,通过在特征空间进行无监督学习获得旋转对称特征簇,统计各个对称特征簇在不同半径尺度下的空间分布来检测图像中旋转对称物体的位置。实验结果表明,本文算法能够对图像中单个、多个旋转对称目标进行精确检测和定位。与传统方法相比,由于该算法建立在对大量关键点进行特征学习的基础上,因此能够有效解决目前对于非规则旋转对称目标检测准确率低的问题。

未来工作主要集中在如何提高算法的执行效率。从复杂度分析可以看到,算法目前的瓶颈主要集中在特征簇的投票阶段,如何通过更有效的特征学习获得更为合理的旋转特征簇并提高投票的效率是重点需要解决的问题。另外,还将使用执行效率更高、更健壮的描述子,进而分析这些描述子对所提出模型的影响,达到提高算法执行效率和准确率的目的。

参考文献

- [1] 吴 刚,杨敬安,李道伦,等. 平面图像的对称性检测方法研究[J]. 计算机科学,2001,28(5):101-103.
- [2] 史 川. 基于图像的三维模型对称性检测方法的研究[D]. 桂林:广西师范大学,2010.
- [3] 肖志涛,史文静,耿 磊,等. 基于相位信息和主成分分析的对称性检测方法[J]. 电子与信息学,2014,36(9):2041-2046.
- [4] Reisfeld D, Wolfson H, Yeshurun Y. Context Free Attentional Operators; The Generalized Symmetry Transform[J]. International Journal of Computer Vision, 1995,14(2):119-130.
- [5] Zabrodsky H, Peleg S, Avnir D. Symmetry as a Continuous Feature[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 1995, 17(12):1154-1166.
- [6] Cornelius H, Loy G. Detecting Bilateral Symmetry in Perspective[C]//Proceedings of Conference on Computer Vision and Pattern Recognition. Washington D. C., USA:IEEE Press,2006:191-191.
- [7] Liu Yanxi, Hays J, Xu Yingqing, et al. Digital Paper-cutting[EB/OL]. (2015-01-13). <http://www.designer-daily.com/digital-papercuts-50295>.
- [8] Prasad V S N, Davis L S. Detecting Rotational Symmetries[C]//Proceedings of the 10th IEEE International Conference on Computer Vision. Washington D. C., USA: IEEE Press,2005:954-961.
- [9] Derrode S, Ghorbel F. Shape Analysis and Symmetry Detection in Gray-level Objects Using the Analytical Fourier-Mellin Representation[J]. Signal Processing, 2004,84(9):25-39.
- [10] Keller Y, Shkolnisky Y. A Signal Processing Approach to Symmetry Detection[J]. IEEE Transactions on Image Processing, 2006,15(8):2198-2207.
- [11] Lee S, Liu Yanxi. Skewed Rotation Symmetry Group Detection[J]. IEEE Transactions on Pattern Analysis & Machine Intelligence, 2010,32(9):1659-1672.
- [12] 姚 会. 基于隐式形状模型的行人检测技术研究[D]. 厦门:厦门大学,2009.
- [13] Lim J J, Zitnick C L, Dollar P. Sketch Tokens: A Learned Mid-level Representation for Contour and Object Detection[C]//Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition. Washington D. C., USA: IEEE Press, 2013: 3158-3165.
- [14] 张丽丽,罗 斌,汤 进,等. 一种基于加权投票的图像匹配改进方法[J]. 计算机工程,2013,39(6):34-37.
- [15] Walther D, Koch C. 2006 Special Issue: Modeling Attention to Salient Proto-objects[J]. Neural Networks, 2006,19(9):1395-1407.

编辑 顾逸斐

(上接第 260 页)

- [5] Sun Fengjie, Wang He, Fan Jieqing. 2D Otsu Segmentation Algorithm Based on Simulated Annealing Genetic Algorithm for Iced-cable Images[C]//Proceedings of International Forum on Information Technology and Applications. Washington D. C., USA: IEEE Computer Society, 2009:600-602.
- [6] 郝颖明,朱 枫. 2 维 Otsu 自适应阈值的快速算法[J]. 中国图象图形学报,2005,10(4):484-488.
- [7] Gonzalez R C, Woods R E, Masters B R. Digital Image Processing[J]. Journal of Biomedical Optics, 2009, 14(2):331-333.
- [8] 吴一全,潘 喆,吴文怡. 二维直方图区域斜分阈值分割及快速递推算法[J]. 通信学报,2008,29(4):77-83.
- [9] Xu Xiangyang, Xu Shengzhou, Jin Lianghai, et al. Characteristic Analysis of Otsu Threshold and Its Applications[J]. Pattern Recognition Letters, 2011, 32(7):956-961.
- [10] 何志勇,孙立宁,黄伟国,等. 基于 Otsu 准则和直线截距直方图的阈值分割[J]. 光学精密工程,2012, 20(10):2315-2323.
- [11] Zhu Ningbo, Wang Gang, Yang Gaobo, et al. A Fast 2D Otsu Thresholding Algorithm Based on Improved Histogram[C]//Proceedings of CCPR'09. Nanjing, China:IEEE Computer Society, 2009:1-5.
- [12] Gong Jian, Li Liyuan, Chen Weinan. Fast Recursive Algorithm for Two-dimensional Thresholding[J]. Pattern Recognition, 1998,31(3):295-300.
- [13] Zhang Yujin. A Survey on Evaluation Methods for Image Segmentation[J]. Pattern Recognition, 1996,29(8):1335-1346.
- [14] Ramac L C, Varshney P K. Image Thresholding Based on Ali-Silvey Distance Measures[J]. Pattern Recognition, 1997,30(7):1161-1174.
- [15] Horng M. Multilevel Thresholding Selection Based on the Artificial Bee Colony Algorithm for Image Segmentation[J]. Expert Systems with Applications, 2011,38(11): 13785-13791.

编辑 刘 冰