

基于图论的分割算法研究综述^{*}

陈 杏¹ 李 军²

(1. 西安工业大学电子信息工程学院 西安 710021)(2. 西安工业大学计算机科学与工程学院 西安 710021)

摘 要 图像分割是计算机视觉领域中的经典问题之一,基于图论的分割算法因其良好的时间性能与分割效果成为众研究者新的关注点。论文在对图像分割算法综述的基础上,重点介绍了基于图论的分割算法并且通过实际仿真分别从时间性能与分割效果上对 Grab Cut 与 One Cut 算法进行了比较。最后对当前该领域存在的问题与发展方向进行了总结。

关键词 图像分割; 能量优化; 迭代最小化; 最优二元分割

中图分类号 TP391.41 DOI:10.3969/j.issn.1672-9722.2016.10.038

Summary of Segmentation Algorithm Based on Graph Theory

CHEN Xing¹ LI Jun²

(1. School of Electronic Information Engineering, Xi'an Technological University, Xi'an 710021)

(2. School of Computer Science and Engineering, Xi'an Technological University, Xi'an 710021)

Abstract In the field of computer vision, image segmentation is one of the classical problems. Because of its good time performance and segmentation results, the segmentation algorithm based on graph theory has become a new focus for the researchers. On the basis of the summary of image segmentation algorithm, this paper focuses on the segmentation algorithm based on graph theory and compares the Grab Cut and One Cut algorithms from the time performance and the segmentation results. Finally, the existing problems and the development direction of this field are summarized.

Key Words image segmentation, energy optimization, iterative minimization, optimal two partition

Class Number TP391.41

1 引言

所谓图像分割指的是根据灰度、颜色、纹理和形状等特征把图像划分成若干互不交迭的区域,并使这些特征在同一区域内呈现出相似性,而在不同区域间呈现出明显的差异性^[1]。图像分割不仅是由图像处理到图像分析的关键步骤,而且是计算机视觉的一种基本技术。若想实现目标特征提取与参数测量,图像分割是其必须经历的步骤,也使得更深层次的图像分析和理解成为可能。针对图像分割领域的研究已经出现了很多算法,但是近些年来,基于图论的图像分割已经成为该领域研究的热点与方向。基于图论的图像分割算法将图像分割映射为加权图的形式,其像素及其颜色、灰度等特征信息分别代表图中的顶点及其属性,像素之间的

邻接关系对应于图的边集,边的权重显示出对应像素的差异度。

图像处理技术不仅在诸多领域扮演着举足轻重的角色,而且图像分割是图像进行处理的键环节之一及基于图论的图像分割理论的前沿性,使针对该类图像分割技术的研究具有较大的价值与应用前景。

2 图像分割方法概述

针对图像分割技术的研究已经经历了很长时间,伴随着计算机技术与机器视觉的不断发展,图像处理技术越来越受到国内外研究者及各行各业的重视。其中,针对图像分割领域中的算法研究也是层出不穷,但至今仍然找不到一个适用于所有类型图像的分割方法,而对于图像分割算法的依据也

^{*} 收稿日期:2016 年 4 月 8 日,修回日期:2016 年 5 月 28 日

作者简介:陈杏,女,硕士研究生,研究方向:计算机视觉。李军,男,硕士研究生,研究方向:智能计算与机器视觉。

是不唯一。本节将对当前主流的图像分割算法的进行归纳与总结,如图 1 所示为图像分割算法分类示意图。



图 1 图像分割算法分类示意图

2.1 基于阈值的分割算法

基于阈值的分割^[2~4]是通过单个或多个阈值将图像的灰度直方图分成几类,认为图像中灰度值在同一灰度级别的像素属于同一物体。阈值分割法先确定适当的分割阈值,然后将图像中所有像素的灰度级与该阈值进行比较,从而可对区域进行划分,达到图像分割的目的。基于阈值的分割算法主要分为三种:基于点、区域的全局及局部与多阈值方法。

2.2 基于边缘的分割算法

基于边缘的图像分割^[5~7]是基于目标对象的边缘灰度变化较大的特点,利用边缘检测的方法将目标对象所属区域进行提取的分割方法。由于该方法是利用图像区域间的不同性质对区域进行划分处理,通常不仅会产生目标对象边缘的断点现象,而且可能产生错误的边缘。按照处理顺序可以将边缘分割技术分为串行与并行边缘检测。

2.3 基于区域的分割算法^[8]

基于区域的分割算法是按照相似性准则对图像进行划分,其方法可归纳为区域生长法、分裂合并法等几种类型。

1) 区域生长法

区域生长法是根据同一物体区域内像素的相似性对像素进行聚集的方法,从初始区域开始,将相同属性的相邻像素或区域归并于当前区域从而达到区域逐步扩散的目的,直至该图像中不存在可以进行归并的像素点或区域。

2) 分裂合并法

分裂合并法首先将图像分成若干互不相交的区域,然后按照相关规则对区域进行分裂或合并从

而实现分割目的,该方法适用于灰度图像与纹理图像分割。

2.4 基于图论的分割算法

基于图论的分割算法将图的最小割问题应用于图像分割领域。该算法首先将图像映射为带权无向图 $G=\langle V, E \rangle$, 图中每个节点 $N \in V$ 对应于图像中的每个像素,每条边 $\in E$ 连接着一对相邻的像素,边的权值表示了相邻像素之间非负相似度(灰度、颜色或纹理)。对图像一个分割 S 相当于一次剪切处理,每个被分割的区域 $C \in S$ 与图中一个子图相对应。基于图论的分割方法的实质是移除特定的边,将图划分为若干子图从而达到分割的目的。目前所了解基于图论的方法有 Graph Cut、Grab Cut 和 One Cut 等。

2.5 基于能量泛函的分割算法

基于能量泛函的分割算法是指活动轮廓模型(Active Contour Model)及其基础上衍生出的相关算法,其基本思想是将目标对象的边缘用连续曲线进行表达,并定义一个能量泛函使得其自变量包括边缘曲线,因此分割过程就转变为求解能量泛函的最小值过程。按照模型中曲线表达形式的不同,活动轮廓模型可以分为两大类:参数活动轮廓模型(Parametric Active Contour model)和几何活动轮廓模型(Geometric Active Contour Model)。

3 基于图论的分割算法

在上文针对图像分割算法分类介绍的基础,本节就基于图论的分割算法展开详细的介绍。依据其算法发展的历程,分别就 Graph Cut、Grab Cut 与 One Cut 算法原理进行深层次地理解与探讨。

3.1 Graph Cut 算法^[9]

在图像处理与机器视觉中,很多问题最后都是汇聚于解决一个最小化能量函数的问题。由于该问题需要在一个相当高维数空间求解一个非凸函数的最小化,以致在此问题的最优化上显的较为困难。Graph Cuts 方法可对一类具有某种特定形式的能量函数得到快速而有效地最优解,导致该方法受到越来越多的关注。20 世纪 90 年代末 Boykov 等将该技术应用于图像分割领域,并提出了该算法。

3.1.1 S-T 图像映射结构

首先用一个无向图 $G=\langle V, E \rangle$ 表示要分割的图像, V 和 E 分别是顶点(vertex)和边(edge)的集合。如图 2 所示为图像转换成图结构后对应的 S-T 图,图中每个顶点对应于图像中每个像素,在此基础上还另外增加了顶点 S 与 T 。S-T 图中存在两

种类型的边,实线表示两个邻域顶点相连形成的边(n-links),虚线表示每个顶点与 S 和 T 相连形成的边(t-links)。在前后景分割中, S 一般表示前景目标, T 一般表示背景。若边集合中的所有边断开将会导致“ S ”与“ T ”图的分开,故将其称为“割”。若一个割的过程中其对应边的所有权值之和最小,就将其称为最小割。

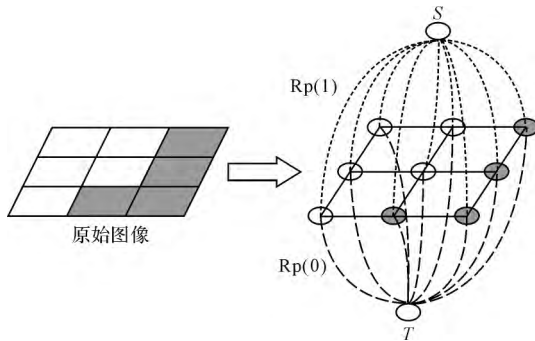


图 2 S-T 图

3.1.2 最小割 S-T 图

由 Boykov 和 Kolmogorov 发明的最大流(Max-Flow)/最小割(Min-Cut)算法可以达到 S-T 图的最小割目的。最小割可以将图的顶点划分为两个没有交集的集合 S 与 T , 其中 $s \in S, t \in T$ 和 $S \cup T = V$ 。此时 S 与 T 分别代表了前景和背景的像素集,同时意味着图像分割过程的结束。

假设整幅图像的标签 label 为 $L = \{l_1, l_2, \dots, l_p\}$, 其中 l_i 为 0(背景)或者 1(目标)。那假设图像的分割为 L 时,图像的能量可以表示为

$$E(L) = aR(L) + B(L)$$

$R(L)$ 为区域项, $B(L)$ 为边界项。 a 为区域项与边界项之间的影响因子,决定了二者对能量的影响程度。 $E(L)$ 为能量函数,图割的目标就是优化该能量函数并使其值最小化。区域项(t-links)中边的权值计算公式为

$$R(L) = \sum_{p \in P} R_p(l_p)$$

$R_p(l_p)$ 能量项的权重可以通过将像素 p 的灰度与给定目标和背景的灰度直方图进行比较而获得,即像素 p 属于标签 l_p 的概率,所以 t-links 的权值如下:

$$R_p(1) = -\ln P_r(l_p | 'obj')$$

$$R_p(0) = -\ln P_r(l_p | 'bkg')$$

边界项:n-links 中每条边权值如下:

$$B(L) = \sum_{(p,q) \in N} B_{\langle p,q \rangle}, \delta(l_p, l_q)$$

其中

$$\delta(l_p, l_q) = \begin{cases} 0 & \text{if } l_p = l_q \\ 1 & \text{if } l_p \neq l_q \end{cases}$$

$$B_{\langle p,q \rangle} \propto \exp\left(-\frac{(l_p - l_q)^2}{2\sigma^2}\right)$$

p 和 q 为相邻像素,边界项体现分割 L 的边界属性, $B_{\langle p,q \rangle}$ 表示像素 p 与 q 之间不连续的程度。

在确定了每条边的权重后,可以通过 min cut 算法对图像进行最小割处理,这些边的断开既实现了对目标与背景的分离,又实现了能量的最小化。

3.2 Grab Cut 算法^[10]

Grab Cut 算法是在 Graph Cut 算法的基础上进行改进而产生的一种新的图像分割算法。该算法利用图像中的纹理与边界信息,只需用户少量的交互操作就可以得到较为满意的分割结果。若目标与背景之间的色差较大将会影响分割的效果。

Grab Cut 在 Graph Cut 算法的基础上做出的改进工作可以归纳为以下几点:

1) Graph Cut 中目标与背景的模型采用灰度直方图,而 Grab Cut 将其改为 RGB 三通道的混合高斯模型(GMM);

2) Graph Cut 中能量最小化是一次性达到,而 Grab Cut 采用交互迭代的方式对其进行分割估计与模型参数的学习;

3) Graph Cut 需要用户指定目标与背景的一些种子像素点,而 Grab Cut 只需提供背景区域的像素集。若需要分割的结果更加精确,可以在初次分割的基础上提供一些确定的种子像素点,在运行该算法即可。

3.2.1 颜色空间模型

Grab Cut 对图像采用 RGB 颜色空间,分别用一个 k 割高斯分量的全协方差混合高斯模型(GMM)对目标及背景进行模型的建立。在该过程中衍生出向量 $K = \{k_1, \dots, k_n, \dots, k_N\}$, 其中 k_n 是第 n 个像素对应的高斯分量, $k_n \in \{1, \dots, K\}$, 则可以将整个图像的 Gibbs 能量定义为

$$E(\underline{\alpha}, k, \underline{\theta}, z) = U(\underline{\alpha}, k, \underline{\theta}, z) + V(\underline{\alpha}, z)$$

结合高斯模型,其中 U 的定义为

$$U(\underline{\alpha}, k, \underline{\theta}, z) = \sum_n D(\alpha_n, k_n, \underline{\theta}, z_n)$$

其中 $D(\alpha_n, k_n, \underline{\theta}, z_n) = -\log p(z_n | \alpha_n, k_n, \underline{\theta}) - \log \pi(\alpha_n, k_n)$, $p(\cdot)$ 与 $\pi(\cdot)$ 分别表示高斯概率分布和混合加权系数,这样 $D(\alpha_n, k_n, \underline{\theta}, z_n)$ 及模型参数 $\underline{\theta}$ 可分别转换成如下形式:

$$D(\alpha_n, k_n, \underline{\theta}, z_n) = -\log \pi(\alpha_n, k_n) + \frac{1}{2} \log \det \sum (\alpha_n, k_n) + \frac{1}{2} [z_n - \mu(\alpha_n, k_n)]^T \sum (\alpha_n, k_n)^{-1} [z_n - \mu(\alpha_n, k_n)]$$

$$\underline{\theta} = \{\pi(\alpha, k), \mu(\alpha, k), \sum (\alpha, k), \alpha = 0, 1, k = \dots, K\}$$

其中 π 表示每一个高斯分量的权重, μ 表示每个高斯分量的均值向量和 Σ 表示每个高斯分量的协方差矩阵, 即目标与背景的混合高斯模型中存在三个参数。在获得某个像素的 RGB 值的基础上, 三个参数的值得以确定, 就可以将其代入目标与背景的 GMM, 就可以得到该像素分别属于目标和背景的概率了, 也就是 Gibbs 能量的区域能量项就可以确定了, 即可以求出图的 t-link 的权值。那么若想得出图像的能量则需要计算出另外一项, 即 $V(\underline{\alpha}, z)$ 。该项表示平滑度, 该项除了对比度之外在单色空间中是基本不变的, 故在颜色空间中根据欧式距离对该项进行计算, 其计算表达式为

$$V(\underline{\alpha}, z) = \gamma \sum_{(m,n) \in C} [\alpha_n \neq \alpha_m] \exp - \beta \|z_m - z_n\|^2$$

参数 β 由图像对比度决定, 如果图像对比度较低, 像素 m 和 n 的差 $\|z_m - z_n\|$ 比较小, 此时需乘一个较大的 β 放大这种差别; 如果图像对比度较高, 像素 m 和 n 的差 $\|z_m - z_n\|$ 比较大, 那么此时需乘以一个较小的 β 缩小这种差别。这样就使得 V 项在对比度高或低的情况下均可正常工作, 此时可以对该图进行分割操作了。

3.2.2 初始化

迭代能量最小化分割算法的初始化过程如下:

1) 通过用户框选目标来得到初始的 trimap T , 可将框外像素视为背景像素 T_B , 而框内像素 T_U 可视为目标的可能像素;

2) 初始化 T_B 内的每一像素 n 的标签 $\alpha_n = 0$; 初始化 T_U 内的每个像素 n 的标签 $\alpha_n = 1$;

3) 经过上述过程, 可以得到目标与背景的一些像素。通过这些像素可进行目标和背景 GMM 的估计。首先通过 k-mean 算法将目标和背景的像素聚类为 K 类, 其次通过 RGB 值估计参数均值和协方差, 最后通过该高斯分量中的像素数目与像素总数的比值确定高斯分量的权重。

3.2.3 迭代最小化

迭代最小化的算法过程如下:

1) 对每个像素分配 GMM 高斯分量: $k_n := \arg \min_{k_n} D_n(\alpha_n, k_n, \theta, z_n)$;

2) 从数据 Z 中学习 GMM 的参数: $\underline{\theta} := \arg \min_{\underline{\theta}} U(\underline{\alpha}, k, \underline{\theta}, z)$;

3) 分割估计: $\min_{\{\alpha_n, n \in T_U\}} \min_k E(\underline{\alpha}, k, \underline{\theta}, z)$;

4) 重复 1) 和 3), 直至收敛;

5) 采用 border matting 对分割的边界进行平滑等等后期处理。

3.3 One Cut 算法^[11]

Tang 等在 Grab Cut 算法的基础上提出了 One Cut 图像分割算法, Grab Cut 算法通过迭代的图切割来达到预期的图像分割目的, 而 One Cut 算法是通过一次图切割便可以得到预期的结果, 并在时间和分割效果上均优于 Grab Cut 算法。

One Cut 算法使用快速全局最优二元分割技术, 明确了最小化目标、背景颜色分布之间的表现重叠, 其一次图切割最小化能量函数表达式如下:

$$E(S) = |\bar{S} \cap R| - \beta \|\theta^S - \theta^{\bar{S}}\|_{L1} + \lambda |\partial S|$$

其中, $R \subseteq \Omega$ 表示包围盒对应的二数码, $S \subseteq \Omega$ 是一个分割段, $l_s = \{s_p \mid p \in \Omega\}$ 是 $S \subseteq \Omega$ 的特征函数。第一项与第二项分别表示包围盒 R 的一个标准膨胀与表现重叠惩罚项; 最后一项是对比敏感平滑项, 其展开式为

$$|\partial S| = \sum w_{pq} |s_p - s_q|$$

$$w_{pq} = \frac{1}{\|p - q\|} \cdot e^{\frac{-\Delta I^2}{2\sigma^2}}$$

$$\sigma^2 = \frac{1}{N_n} \sum_{(p,q) \in n(I)} \|I_p - I_q\|^2, \Delta I^2 = \|I_p - I_q\|^2$$

N_n 指 $n(I)$ 元素个数, $n(I)$ 的值表示图像 I 中相邻像素对集合

$$\beta = \beta_{mg} = \frac{|R|}{-\|\theta^R - \theta^{\bar{R}}\|_{L1} + |\Omega|/2} \cdot \beta'$$

β' 是一个全局参数, 经实验得出 $\beta' = 0.9, \lambda = 9$ 。

用户根据特定图像提供一个其感兴趣的目标矩形包围盒, 矩形包围盒外面的像素使用硬性约束分配为背景, 又可以将分割能量函数简化为以下形式:

$$E_{seeds}(S) = -\beta \|\theta^S - \theta^{\bar{S}}\|_{L1} + \lambda |\partial S|$$

由于 One Cut 算法对矩形框比较敏感, 则需要用户绘制的矩形框要完全覆盖目标, 并且尽可能小。该方法使用了少量的辅助节点实现了高效率的图像分割, 与 Grab Cut 算法相比, 对于复杂图像来说该算法依然具备很好的分割效果与时间性能。

4 实验结果与分析

通过前面针对图论的图像分割算法的详细介绍, 本节就 Grab Cut 与 One Cut 图像分割算法进行分割效果与算法性能的分析。

4.1 Grab Cut 与 One Cut 实验结果

在充分理解 Grab Cut 与 One Cut 的算法原理基础上, 分别对这两个分割算法做了分割效果的实验, 其图像分割结果分别如图 3、4 所示。



图3 Grab Cut 图像分割结果

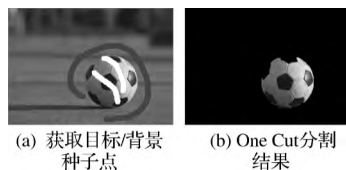


图4 One Cut 图像分割结果

由图3,4比较可知,虽然从分割结果上来说,它们均达到了较好的效果,但是从交互方式上来看,One Cut 算法更具优势。

4.2 Grab Cut 与 One Cut 时间性能分析

本文针对 Grab Cut 与 One Cut 时间性能做了具体而实际地分析工作,它们的时间性能情况如图5中所示。

针对 One Cut 每一次分割过程中,取其时间消耗值,样本数为10。通过对 Grab Cut 递增其迭代次数得到其时间消耗值,迭代次数由1逐渐增至10。由图中可知,Grab Cut 随着迭代次数的增加,其时间消耗越大;One Cut 各个样本的时间变化不大,时间消耗始终在100ms上下浮动。

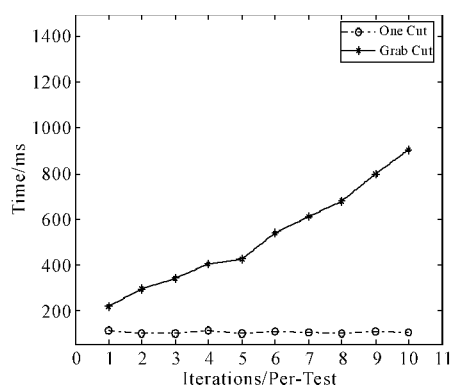


图5 Grab Cut 与 One Cut 算法时间性能对比图

通过实验与分析可知,在相同分割效果的前提下,One Cut 在交互及时间性能上均是优于 Grab Cut 图像分割算法。

5 结语

本文在描述图像分割算法的基础上,重点针对基于图论的分割算法进行算法原理进行详细阐述,并对 Grab Cut 与 One Cut 算法进行分割效果及时间性能上的实验分析。虽然在分割效果上满足要求,但是由于分割过程中对交互的依赖性过大,达

不到图像分割的自动化与实时性的目标,而且在该类方法是基于图像像素为对象展开方法的研究工作,若图像中像素过多将会导致计算量过大。所以在保证分割效果的前提下,算法效率的提高将是图像分割算法在未来的发展过程中很长时间内亟待解决的课题。

参考文献

- [1] Abadi D J, Carney D, Cetintemel U, et al. Aurora: A New Model and Architecture for Data Stream Management[J]. VLDB Journal, 2003, 12(2): 120-139.
- [2] Noma A, Graciano A B V, Consularo L, et al. A new algorithm for interactive structural image segmentation [J]. The Computing Research Repository, 2008(1): 4-6.
- [3] 武红玉. 阈值分割算法在图像处理中的应用[J]. 科技信息, 2012(27): 201-202.
WU Yuhong. Application of threshold segmentation algorithm in image processing[J]. Science and Technology Information, 2012(27): 201-202.
- [4] Han S Q, Wang L. The review of threshold methods in image segmentation[J]. System Engineering and Electronics, 2002, 4(6): 91-102.
- [5] Mortensen E. Interactive segmentation with intelligent scissors[J]. Graphical Models and Image Processing, 1998, 60(5): 349-384.
- [6] Falcao A X, Udupa J K, Miyazawa F K. An ultra-fast user-steered image segmentation paradigm: livewire on the fly [J]. IEEE Transaction on Medical Imaging, 2000, 19(1): 55-62.
- [7] De Miranda P A V, Falcão A X, Udupa J K. Synergistic arc-weight estimation for interactive image segmentation using graphs [J]. Computer Vision and Image Understanding, 2010, 114(1): 85-99.
- [8] 舒添慧, 胥布工, 胡战虎. 基于区域生长法的医学图像分割[J]. 微计算机信息, 2008, 24(6-3): 284-285.
SHU Tianhui, XU Bugong, HU Zhanhu. Medical image segmentation based on region growing method[J]. Microcomputer information, 2008, 24(6-3): 284-285.
- [9] Yuri Boykov M. P. Jolly. Interactive graph cuts for optimal boundary region Segmentation of objects in N-D images[C]//Proceedings of ICCV, 2001: 106-110.
- [10] Carsten Rother, Vladimir Kologorov, Andrew Blake. "GrabCut"—Interactive foreground extraction using iterated graph cuts [J]. In ACM Transactions on Graphics (SIGGRAPH), August 2004: 3-10.
- [11] TANG M, GORELICK L, VEKSLER O, et al. GrabCut in One Cut[C]//International Conference on Computer Vision, Sydney: IEEE, 2013: 1769-1776.