

融合纹理特征的 SEEDS 超像素分割算法

陈 畅, 刘永坚, 解 庆

(武汉理工大学 计算机科学与技术学院, 湖北 武汉 430070)

摘 要: 在 SEEDS 算法的基础上, 进行了改进, 提出了 T-SEEDS 算法. 其通过融合纹理信息, 来增强图像分割的效果, 并较好的解决了超像素个数较小的情况下, SEEDS 算法分割效率不高的问题.

关键词: 超像素分割; 纹理特征; SEEDS

中图分类号: TP391.41

文献标识码: A

文章编号: 1000-7180(2018)03-0064-04

SEEDS-Based Superpixel Segmentation Method With Texture Fusion

CHEN Chang, LIU Yong-jian, XIE Qing

(School of Computer Science and Technology, Wuhan University of Technology, Wuhan 430070, China)

Abstract: This article introduces an improved algorithm based on SEEDS, T-SEEDS. T-SEEDS algorithm enhances the efficiency of image segmentation by add texture information into the algorithm, which solves the problem nicely that SEEDS can't segment effectively with small number of superpixel at the same time.

Key words: superpixel segmentation; texture feature; SEEDS

1 引言

近年来,超像素(superpixel)分割发展迅速,在国内外得到越来越广泛的研究与运用.通常情况下,图像均有数量庞大的像素点,处理图像时会有巨大的计算量,而对图像进行预处理可以有效降低计算复杂度. Ren 和 Malik 提出的超像素概念为图像预处理提供了新的思路^[1].

超像素,就是在图像中将具有一致特征的局部区域,作为图像处理的子单元.相比于普通图像分割方法以像素作为基本单元,超像素大大降低了图像处理的计算复杂度,提升了计算效率,尤其在视频图像分割、3D 图像分割等对实时性要求较高的领域,表现突出,具有很大的研究意义和良好的应用前景^[2].

根据原理的不同,超像素分割大概可分为两种.第一种是基于梯度的超像素分割算法.具有代表性的算法有 Mean Shift 算法^[3]、Turbo pixel 算法^[4]、经典的分水岭算法,以及 SLIC(Simple Linear Iterative Clustering)算法^[5],第二种是基于图论的超像

素分割算法.有 N-Cuts(Normalized cuts)算法^[6]、Superpixel Lattices(简称 SL)算法^[7]及 GCa 和 GCb 算法等^[8].

SEEDS 算法(Superpixels Extracted via Energy-Driven Sampling)也是一种基于图论的超像素分割算法^[9].它是当前分割图像表现最好超像素分割算法之一,但是 SEEDS 算法只利用了图像的颜色信息进行分割,在某些情况下,不能很好地贴合目标轮廓,降低了分割精度.本文通过在 SEEDS 算法中结合图像的纹理信息进行超像素分割,获得了良好的结果.

2 SEEDS 超像素分割

SEEDS 算法^[9]不同于传统方法,其通过一个规则的网格分割图像,可以将这个过程看作初步的超像素分割,网格就是最初的超像素.然后逐步移动当前超像素的边缘,使得超像素内部的像素拥有尽可能统一的颜色特征,来快速完成分割.

SEEDS 定义了一个能量函数作为目标函数,来度量每个超像素的值.如式(1)所示.

$$E(s) = H(s) + \gamma G(s) \quad (1)$$

式中, $H(s)$ 是颜色表达式, 表示超像素的颜色信息. 超像素中颜色越单一, 颜色分布越集中, $H(s)$ 值越大; $G(s)$ 是形状表达式, 用来表示超像素的规整程度; γ 为系数, 用来调整颜色表达式与形状表达式的权重. SEEDS 算法具体步骤如下所示.

(1) 使用 K 个网格对图像进行预分割. K 即为最终分割出的超像素的个数; K 的大小可人为设定. 规定第 k 个超像素区域用 A_k 表示.

(2) 计算颜色表达项 $H(s)$. 首先为每个超像素区域建立一个颜色直方图, 将颜色空间划分为若干区域, 每个区域在直方图中用一个箱格表示, 记为 H_j . 定义 $C_{A_k}(j)$ 作为 A_k 中像素在颜色直方图中的分布. 如式(2)所示.

$$C_{A_k}(j) = \frac{1}{Z} \sum_{i \in A_k} \delta(I(i) \in H_j) \quad (2)$$

式中, $I(i)$ 代表像素 i 的颜色; Z 是一个规范化因子, 用于将公式归一化; $\delta(\cdot)$ 是一个指示函数, 当像素 i 的颜色落在颜色直方图的箱格 j 中时, $\delta(\cdot)$ 值取 1. 再定义一个函数 $\psi(C_{A_k})$ 用来衡量每个超像素区域中的颜色分布. 如式(3)所示.

$$\psi(C_{A_k}) = \sum_{H_j} (C_{A_k}(j))^2 \quad (3)$$

当超像素区域中的像素在颜色直方图中越集中, $\psi(C_{A_k})$ 取值越大, 当所有的像素都集中在颜色直方图的同一个箱格上时, $\psi(C_{A_k})$ 取值达到最大, 为 1. 在此基础上, 可以得到能量公式的颜色表达项 $H(s)$, 即所有超像素颜色项的和, 如式(4)所示.

$$H(s) = \sum_k \psi(C_{A_k}) \quad (4)$$

(3) 计算边缘表达项 $G(s)$. $G(s)$ 通过惩罚超像素边缘之间的局部不规整性, 来帮助产生贴合目标轮廓的边缘. 首先用 $N \times N$ 的网格分割图像, 然后为每个网格建立直方图. 每个直方图中有 K 个箱格, 代表着 K 个超像素. 然后统计每个网格中的像素在直方图中的分布, 定义 N_i 表示像素包含 i 的网格, 则直方图可以用式(5)表示.

$$B_{N_i}(k) = \frac{1}{Z} \sum_{j \in N_i} \delta(I(i) \in A_k) \quad (5)$$

为了使得超像素拥有更好、更平滑的边缘, 需要只包含一个超像素的网格尽可能多. 所以用式(6)定义边缘项 $G(s)$:

$$G(s) = \sum_i \sum_k (B_{N_i}(k))^2 \quad (6)$$

当 N_i 中的像素只属于一个超像素时, $G(s)$ 取值达到最大.

(4) 迭代的实现超像素分割. 在每次迭代中选取能量最低的超像素, 将边缘附近的像素移入邻居超像素, 使颜色分布更集中. 最后设定一个时间 t_{stop} 作为算法运行的终点, 时间到达 t_{stop} 时中止算法的运行.

3 纹理特征分析

纹理是一种重要的图像特征. 在部分图像中, 纹理可以定义区域的特性, 对获得正确的分析具有重要的作用. 关于纹理的定义, 主要有两种方法, 即结构方法和统计方法. 结构方法是将纹理看作具有某种规则或重复关系的基本纹理素(texel)的集合. 统计方法则将纹理看作是在一个区域中亮度值分布的一种定量度方法. 相比于结构方法, 统计方法更易于计算, 鲁棒性更好, 在实际中更常用. 常用的统计方法有 LPB(局部二值分解)纹理测度、共生矩阵测度、Laws 纹理能量测度、自相关与功率谱等.

通常将边缘检测作为纹理分析的第一步, 边缘的密度与方向是重要的纹理特征. 在固定大小的区域内, 边缘像素点的个数反映了该区域的纹理分布密度. 统计边缘的方向也有助于刻画纹理模式. 在一个含有 N 个像素的区域 R 内, 对该区域应用基于梯度的边缘检测算子, 对每个像素 p 产生两个输出: 梯度幅值 $\text{Mag}(p)$ 和梯度方向 $\text{Dir}(p)$. 然后为梯度幅值与梯度方向建立规范化直方图, 分别用 $H_{\text{mag}}(R)$ 与 $H_{\text{dir}}(R)$ 表示. 则可用式(7)对区域 R 中的纹理进行定量描述:

$$\text{Fedge} = (H_{\text{mag}}(R), H_{\text{dir}}(R)) \quad (7)$$

例如 R 区域有 25 个像素, 每个像素内都有一条边缘, 其中 5 条暗边缘, 20 条亮边缘, 水平、垂直、对角方向上分别有 6, 8, 11 条边缘, 则其单位面积的边缘数就是 1.0, 规范化梯度幅值直方图为 $\{0.2, 0.8\}$, 规范化梯度方向直方图为 $\{0.24, 0.32, 0.44\}$.

此外, 可通过计算直方图的 L_1 距离, 对两幅图像进行比较, 若直方图分别为 H_1 和 H_2 , 直方图的箱格数为 n , 则计算公式如式(8)所示.

$$L_1(H_1, H_2) = \sum_{i=1}^n |H_1[i] - H_2[i]| \quad (8)$$

4 T-SEEDS 超像素分割算法

4.1 T-SEEDS 算法流程

T-SEEDS 算法主要是在 SEEDS 算法的基础上融合纹理信息进行图像分割. 首先用 K 个网格对图

像进行预分割,然后为每个网格分别建立颜色与纹理直方图.实验表明,在 LAB 颜色空间中 SEEDS 算法分割效果最优^[9],因而在 LAB 空间中建立颜色直方图,将 LAB 空间的每个通道分为几个区段后随机组合,作为直方图的分量.纹理直方图的构建过程将在本章最后一节的内容中介绍.统计每个网格中的像素在颜色与纹理直方图中的分布,带入新的能量公式计算结果,再通过迭代的移动网格边界实现能量方程数值最大化,使颜色与纹理特征相同的区域成为一个超像素,完成分割. T-SEEDS 算法流程如图 1 所示.

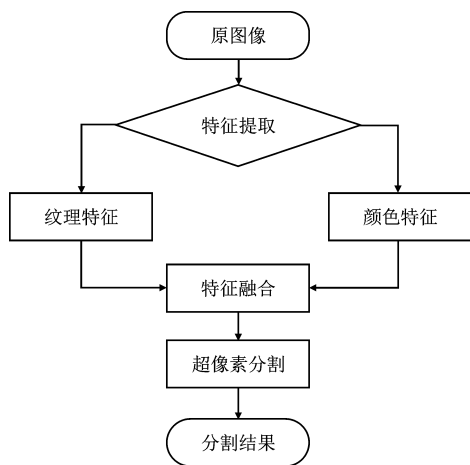


图 1 T-SEEDS 分割算法流程图

4.2 纹理信息提取

为了方便与 SEEDS 算法通过颜色直方图计算能量的方法进行结合,本文采用 Laws 纹理能量测度方法^[10]来提取图像纹理特征.

Laws 纹理能量方法首先在一个固定大小的窗口中,使用模板来检测不同类型的局部纹理.其通过使用不同的卷积模板对图形进行卷积,提取不同类型的纹理信息,并计算出图像的能量值,输出对应的能量图.模板通过模板向量算出,向量表示如下:

$$L_5(\text{水平}) = [1 \ 4 \ 6 \ 4 \ 1]$$

$$E_5(\text{边缘}) = [-1 \ -2 \ 0 \ 2 \ 1]$$

$$L_5(\text{点}) = [-1 \ 0 \ 2 \ 0 \ -1]$$

$$W_5(\text{波纹}) = [1 \ -4 \ 6 \ -4 \ 1]$$

如上述表达式所示, L_5 向量计算图像的局部均值, E_5 向量用于计算图像的边缘特征, S_5 向量用来检测点, R_5 向量用来检测波纹.通过计算一对向量的外积来得到 2D 卷积模板,例如,模板 E_5L_5 是通过计算 E_5 、 L_5 的乘积得到的.

Laws 方法首先使用去均值法获得预处理的图像来去除光照干扰,然后用模板对预处理后的图像

进行滤波.四种模板向量两两相乘,得到 16 个 5×5 的卷积模板,对图像进行卷积后得到 16 个处理结果.若将像素 $[i, j]$ 使用第 k 个模板计算后得到的结果记为 $F_k[i, j]$,那么滤波器 k 对应的纹理能量图 E_k 可表示为

$$E_k[r, c] = \sum_{j=c-7}^{c+7} \sum_{i=r-7}^{r+7} |F_k[i, j]| \quad (9)$$

在产生的 16 幅能量图中,有一些向量对是对称的,可以相互组合,用他们的平均值代替,最终简化为 9 个图.例如, $E5L5$ 、 $L5E5$ 可分别用来检测水平和垂直边缘.通过计算两种模板滤波后产生的能量图的平均值可以检测总边缘.这样,对每一幅目标图像处理后都输出 9 个能量图,代表着这幅图像不同的纹理特征.

4.3 融合纹理特征的超像素分割

为了融合图像的颜色和纹理特征,将建立纹理直方图,并修改式(1)所展示的能量公式.在上一节的内容中,介绍了 T-SEEDS 提取纹理特征的方法,最终会产生 9 个能量图,代表着 9 种不同类型的纹理特征.因为同一个超像素区域中,图像应具有相同或相似的纹理特征,所以使用与第三章中介绍的纹理分析方法类似的方法建立纹理直方图,并将这 9 种纹理特征作为直方图的箱格,然后统计每个网格中的像素在直方图中的分布情况,将纹理相同或相近的区域分割为一个超像素.使用 $T_{A_k}(j)$ 表示每个箱格中像素的分布. $T_{A_k}(j)$ 的表示与颜色直方图分布 $C_{A_k}(j)$ 类似,如式(10)所示.

$$T_{A_k}(j) = \frac{1}{Z} \sum_{i \in A_k} \delta(I(i) \in H_j) \quad (10)$$

则新的能量方程如式(11)所示:

$$\psi_{\text{total}} = \sum [\gamma * (C_{A_k}(j))^2 + (1 - \gamma) * (T_{A_k}(j))^2] \quad (11)$$

式中, γ 为为一个固定的参数,用来调节颜色信息与纹理信息在能量方程中的比例,在本文的实验中 γ 取值为 0.4.

为了均衡分割质量与性能,需要设置适当的颜色直方图箱格的数量与用于分割的网格数量 K .在建立颜色直方图时,将 LAB 空间的每个通道,分为 5 个区段,这样就会在颜色直方图中产生 $5 \times 5 \times 5$ 共 125 个箱格用于目标图像的像素分类.在实验中,将会比较不同数量的超像素对最终分割结果的质量与效率的影响,来最终确定超像素个数 K .

5 实验与分析

在本文的实验中,将在 BSDS500 图像库上进行

验证试验. 通过在 200 幅自然图像上分别使用 SEEDS 算法与 T-SEEDS 算法进行不同粒度的超像素分割后进行比较, 实验中设置的超像素个数分别为 10, 15, 20, 50, 100, 150, 200. 在实验结果分析上, 将使用边缘召回率和欠分割错误率两种主流标准来评价分割后的边缘贴合程度, 通过计算 50 次迭代后 SEEDS 算法与 T-SEEDS 算法的运行时间来比较两种算法的计算效率. 本实验的实验结果均在内存为 8.00 GB, 处理器为 Intel (R) Core (TM) i5-7400 CPU @3.00 GHz 的计算机上得出.

在超像素个数设定为 50 的情况下, 将 SEEDS 算法与 T-SEEDS 算法对自然场景图片分割结果的对比后发现, 相比于 SEEDS 算法, T-SEEDS 算法得出的超像素大小更均匀, 边缘更规整, 具有更好的贴合度. 通过实验与统计, 发现 T-SEEDS 边缘召回率与欠分割错误率均优于 SEEDS 算法, 这与观察得出的结论相吻合.

使用上述两种算法对实验图像进行超像素分割实验, 超像素个数分别取 10, 15, 20, 50, 100, 150, 200 等, 然后在这 7 个等级上对两种算法的分割效果以及运行效率进行对比. 两种算法在每个等级上的运行结果, 是对 200 幅实验图像分割后取算数平均数得到的实验结果, 具有较好的代表性. 图 2 和图 3 分别展示了两两种算法的欠分割错误率以及边缘召回率的比较.

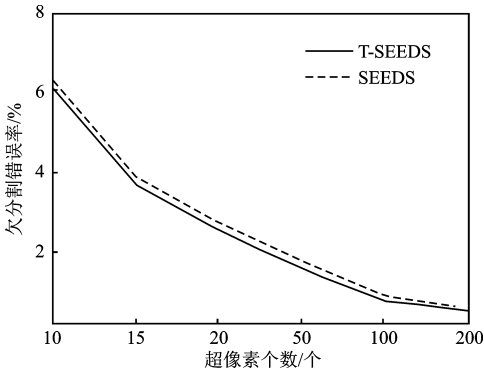


图 2 欠分割错误率

从图 2 和图 3 中发现, T-SEEDS 算法的边缘贴合度更好. T-SEEDS 算法的欠分割错误率要小于原生的 SEEDS 算法, 而边缘召回率在小于 50 个超像素时明显高于 SEEDS 算法, 超像素个数较多时, 两种算法的召回率基本持平.

在运行效率方面, 通过比较 SEEDS 算法与 T-SEEDS 算法 50 次迭代所需要的时间长短来评估算法效率. 如图 4 所示, 因为在每次迭代中, T-SEEDS

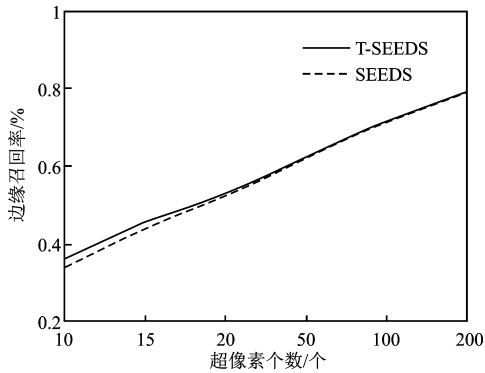


图 3 边缘召回率

算法增加了纹理直方图的处理, 所以时间要略长于 SEEDS 算法, 然而直方图的更新操作很快, 因而时间长度差别不大, 特别是在超像素个数低于 50 时, 两种算法的运行时间基本相同. 此外, 实验发现在超像素个数为 50 时, 分割效果较好, 效率较高, 是在分割质量与效率间一个良好的折衷.

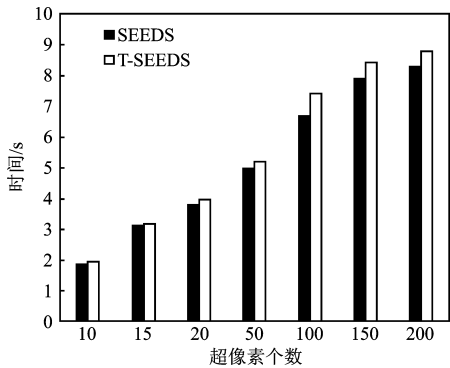


图 4 50 次迭代运行时间对比图

综上所述, T-SEEDS 算法在保证算法运行效率的同时, 较好地提高了 SEEDS 算法的边缘贴合度, 增强了算法的图像分割效果, 是对 SEEDS 算法很好的改进.

6 结束语

超像素分割是现在广泛使用的图像预处理技术, 其目标是能快速的分割出符合人类认知的目标区域边界, 降低后续处理图像的计算复杂度. 本文通过研究 SEEDS 算法, 提出了一种改进的算法 T-SEEDS 算法. 实验证明, T-SEEDS 算法通过融合图像的纹理与颜色信息, 在保证算法优秀的分割效率的基础上, 获得更贴合目标轮廓、对复杂纹理适应性更好的分割结果, 特别是解决了原算法在超像素个数较少时, 分割精度不高的问题, 具有一定的现实意义

(下转第 73 页)

- cessing, 2013, 22(2): 573-584.
- [2] 肖林云,陈秀宏,林喜兰. 邻域信息熵的核模糊C均值聚类图像分割算法[J]. 微电子学与计算机, 2016, 33(11):166-170.
 - [3] Das S, Sil S. Kernel-induced fuzzy clustering of image pixels with an improved differential evolution algorithm [J]. Information Sciences, 2010, 180(8): 1237-1256.
 - [4] Bezdek JC. Pattern recognition with fuzzy objective function algorithms[M]. Springer, 1981.
 - [5] Ahmed M N, Yamany S M, Mohamed N, et al. A modified fuzzy c-means algorithm for bias field estimation and segmentation of MRI data[J]. IEEE Transactions on Medical Imaging, 2002, 21(3):193-199.
 - [6] Chen S, Zhang D. Robust image segmentation using FCM with spatial constraints based on new kernel-induced distance measure [J]. IEEE Transactions on Systems Man & Cybernetics Part B Cybernetics, 2004, 34(4):1907-1916.
 - [7] Szilágyi L, BenyóZ, Szilágyi S M, et al. MR brain image segmentation using an enhanced fuzzy c-means algorithm[C] // Annual International Conference of the IEEE Engineering in Medicine and Biology Society. IEEE, 2003: 17-21.
 - [8] Cai W, Chen S, Zhang D. Fast and robust fuzzy c-means clustering algorithms incorporating local information for image segmentation[J]. Pattern Recognition, 2007, 40(3): 825-838.
 - [9] Krinidis S, Chatzis V. A robust fuzzy local information c-means clustering algorithm[J]. IEEE Transactions on Image Processing, 2010, 19(5): 1328-1337.
 - [10] 王保平, 范九伦, 谢维信, 等. 一种基于代价函数和模糊熵的图像分割方法[J]. 光子学报, 2003, 32(12): 1502-1505.
 - [11] Benaichouche A N, Oulhadj H, Siarry P. Improved spatial fuzzy c-means clustering for image segmentation using PSO initialization, Mahalanobis distance and post-segmentation correction [J]. Digital Signal Processing, 2013, 23(5):1390-1400.

作者简介:

朱占龙 男, (1984-), 博士, 讲师. 研究方向为图像分割.

E-mail: zzl_seu@163.com.

郑一博 男, (1981-), 博士, 副教授. 研究方向为图像处理、人工智能.

王远 男, (1979-), 博士, 讲师. 研究方向为光纤光栅传感.

(上接第67页)

义与实用价值. 下一步将继续在融合颜色与纹理信息的基础上, 调节能量方程中两种特征的权值, 使得T-SEEDS算法在超像素数量较多的情况下, 也能获得良好的改进, 更好的图像分割效果.

参考文献:

- [1] Ren X F, Malik J. Learning a classification model for segmentation[C] // 9th IEEE International Conference on Computer Vision. France, 2003: 10-17.
- [2] 宋熙煜, 周利莉, 李中国, 等. 图像分割中的超像素方法研究综述[J]. 中国图像图形学报, 2015, 20(5): 599-608.
- [3] Comaniciu D, Meer P. Mean shift: a robust approach toward feature space analysis[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2002, 24(5): 603-619.
- [4] Levinshtein A, Stere A, Kutulakos K N, et al. TurboPixels: fast superpixels using geometric flows[J]. IEEE Transactions on Pattern Analysis & Machine Intelligence, 2009, 31(12):2290-2297.
- [5] Achanta R, Shaji A, Smith K, et al. Slic superpixels [R]. Lau-sanne, Vaud, Switzerland: Swiss federal Institute of Technology, 2010.
- [6] Shi J, Malik J. Normalized cuts and image segmentation [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2000, 22(8): 888-905.
- [7] Moore A P, Prince S J D, Warrell J, et al. Superpixel lattices[C] // Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on. IEEE, 2008:1-8.
- [8] Veksler O, Boykov Y, Mehrani P. Superpixels and supervoxels in an energy optimization framework[C] // Computer Vision ECCV 2010, European Conference on Computer Vision, Heraklion, Crete, Greece, September 5-11, 2010, Proceedings. DBLP, 2010:211-224.
- [9] Van den Bergh M, Boix X, Roig G, et al. SEEDS: superpixels extracted via energy-driven sampling [J]. International Journal of Computer Vision, 2012, 111(3): 298-314.
- [10] Boix X, Roig G, Leistner C, et al. Nested Sparse Quantization for Efficient Feature Coding[C] // European Conference on Computer Vision. Springer-Verlag, 2012:744-758.

作者简介:

陈畅 男, (1994-), 硕士研究生. 研究方向为计算机视觉、机器学习、人工智能. E-mail: 1344683503@qq.com.

刘永坚 男, (1962-), 博士, 教授, 博士生导师. 研究方向为数字出版、数字传播、知识服务.

解庆 男, (1986-), 博士, 副教授, 硕士生导师. 研究方向为流数据挖掘与模式分析、连续查询优化、知识服务与推荐系统.