超像素分割算法研究综述*

王春瑶,陈俊周[†],李 炜 (西南交通大学 信息科学与技术学院,成都 610031)

摘 要:超像素能够捕获图像冗余信息,降低后续处理任务复杂度,已受到了国内外研究者的日益关注。首先分析了超像素分割领域的发展现状,以基于图论的方法和基于梯度下降的方法为视角,对现有超像素分割方法进行归纳和论述。在此基础上,就目前常用的超像素分割算法进行了实验对比,分析各自的优势和不足。最后,对超像素分割技术的最新应用进行了介绍和展望。

关键词:超像素;图像分割;图论;梯度下降

中图分类号: TP391.4 文献标志码: A

doi:10.3969/j.issn.1001-3695.2014.01.002

文章编号: 1001-3695(2014)01-0006-07

Review on superpixel segmentation algorithms

WANG Chun-yao, CHEN Jun-zhou[†], LI Wei

(College of Information Science & Technology, Southwest Jiaotong University, Chengdu 610031, China)

Abstract: Superpixel can capture redundancy of the image and reduce the complexity of subsequent processing tasks. These advantages make it receive more and more attentions from researchers at home and abroad. This paper first analyzed the development of the superpixel segmentation, and summarized the state-of-the-art superpixel segmentation algorithms in the view of graph-based and gradient-ascent-based methods. Then, it compared several superpixel segmentation algorithms by experiments, and illustrated their strengths and weaknesses respectively. At last, it introduced the latest applications of superpixel segmentation techniques with prospects.

Key words: superpixel; image segmentation; graph; gradient-descent

0 引言

图像分割是指按照一定的相似性准则将图像划分成具有特殊语义的不同区域,其研究最早可以追溯至 20 世纪 60 年代,已历经几十年的发展。图像分割作为计算机视觉领域的基本问题,是图像理解的重要组成部分。与此同时,它在图像处理、模式识别和人工智能等多个领域也扮演了关键的角色。

目前对图像的处理大多以像素为单位,用二维矩阵来表示一张图像,并未考虑像素之间的空间组织关系,这使得算法处理效率过低。2003 年,Ren 等人^[1]最早提出了超像素这一概念,所谓超像素,是指具有相似纹理、颜色、亮度等特征的相邻像素构成的图像块。它利用像素之间特征的相似程度将像素分组,可以获取图像的冗余信息,在很大程度上降低了后续图像处理任务的复杂度。

超像素生成算法大致可分为基于图论的方法和基于梯度下降的方法两类。具有代表性的基于图论的超像素分割方法有 Felzenswalb 等人^[2]提出的 graph-based 方法、Shi 等人^[3,4]提出的 Ncut(normalized cuts)方法、Moore 等人^[5]的 superpixel lattice 方法和 Liu 等人^[6]提出的基于熵率方法。其中,Ncut 算法利用轮廓特征和纹理特征来全局最小化代价函数,它能生成规则的超像素,但是图像边界的保持效果不好、计算量较大,处理大图片时速度很慢。Graph-based 方法使用最小生成树的思想

来分割图像,它能较好地保持图像边界,速度较快,但是得到的超像素大小和形状都不规则。Superpixel lattice 方法保持了图像的拓扑结构信息,但性能严重依赖于预先提取的图像的边界。熵率法提出了包括图像随机游走熵率和平衡项的目标函数,通过最大化目标函数以实现分割,它产生的超像素比较规则和均匀。

对于基于梯度下降法,已有的研究有 Vincent 等人^[7]的分水岭 (watersheds) 方法、Comaniciu 等人^[8]的 MeanShift 方法、Vedaldi 等人^[9]的 Quick-shift 方法、Levinshtein 等人^[10]的 Turbopixels 方法和 Achanta 等人^[11,12]的 SLIC (simple linear iterative clustering) 方法。它们都采用了聚类的基本思想,但各的自具体方法不同,也有不同的优缺点。Turbopixels 是基于几何流的水平集方法,对初始的种子点逐步进行碰撞,最终将超像素近似均匀地分布到图像平面上。Mean-Shift 是一个迭代模态搜索的过程,它能产生形状规则的超像素,但速度慢,并且不能控制超像素的数量、大小和紧凑度。Watersheds 是一种基于拓扑理论的数学形态学分割方法,该方法速度快,但是不能控制超像素的个数和紧凑度。SLIC 算法是基于颜色和距离相似性进行超像素分割,该方法思想简单,可以生产大小均匀、形状规则的超像素。

由于超像素能够提取中层图像特征,它通常作为分割算法的预处理步骤,已经成为视觉领域的一项关键技术。此外,许

收稿日期: 2013-05-29; **修回日期**: 2013-07-03 **基金项目**: 国家自然科学基金资助项目(61003143);中央高校基本科研业务费专项资金 资助项目(SWJTU12CX094)

作者简介:王春瑶(1989-),女,硕士研究生,主要研究方向为超像素分割;陈俊周(1979-),男(通信作者),副教授,硕导,博士,主要研究方向为 计算机视觉、模式识别、数字图像处理(jzchen@ swjtu. edu. cn);李炜(1987-),男,硕士研究生,主要研究方向为数字图像处理. 多的研究工作表明,超像素技术已成功应用于多种视觉任务,如骨架提取^[13]、图像分割^[14-16]、人体姿势估计^[17,18]、目标识别^[19,20]、目标跟踪^[21-24]等,因此它在计算机视觉应用中也越来越受到研究者的重视,具有很大的发展潜能和应用价值。本文对超像素分割算法,特别是近几年来出现的新思路、新方法给出了具体的介绍和分析,并在文章的最后对超像素的应用进行了展望。

1 基于图论的超像素分割方法

基于图论的图像分割是一种自上而下的全局分割方法,其 主要思想是把整幅图像看做一幅带权无向图,图像中每一个像 素对应图中的一个节点,像素之间的相邻关系对应图的边,像 素特征之间的差异或相似性对应边上的权重。然后在所建立 的图上利用各种分割准则来对图中的节点进行划分,进而完成 对图像的分割。

1.1 Graph-based 方法

Felzenszwalb 等人 $^{[2]}$ 提出的基于图论分割方法采用了最小生成树的思想,目的是使同一区域内的元素尽可能相似,不同区域的元素尽可能不相似。该方法的思想是将一幅图像映射为无向图时,为每条边 e 定义了权值 w(e) ,它用来表示边 e 所连接的两个顶点 v_i 和 v_j 的差异性。V 是图中所有顶点的集合,一个分割 S 就是将 V 分成不同区域的一种划分,每个小区域 $C \in S$ 对应于图 G' = (V, E') 的一个连通子图,其中 E'是 E 的非空子集。

对于子集 $C \subseteq V$ 的内部差异就是该区域最小生成树 MST (C,E) 上的最大权值,计算公式为

$$int(C) = \max_{e \in MST(C, E)} w(e) \tag{1}$$

两部分子集 C_1 , $C_2 \subseteq V$ 的区域间差异为连接这两部分的最小权值边,即

$$\operatorname{dif}(C_{1}, C_{2}) = \min_{v_{i} \in C_{1}, v_{j} \in C_{2}, (v_{i}, v_{j}) \in E} w((v_{i}, v_{j}))$$
 (2)

如果两个部分间的区域差异 $\operatorname{dif}(C_1,C_2)$ 大于 C_1 、 C_2 中至 少一个部分的内部差异 $\operatorname{int}(C_1)$ 或 $\operatorname{int}(C_2)$,则说明这两部分不能合为一体,否则就可以合并,用下面公式来说明:

$$D(C_1, C_2) = \begin{cases} \text{true} & \text{if } \operatorname{dif}(C_1, C_2) > \operatorname{MInt}(C_1, C_2) \\ \text{false} & \text{otherwise} \end{cases} \tag{3}$$

$$\mathsf{MInt}(\,C_1\,,C_2\,) = \min(\,\mathsf{int}(\,C_1\,)\,+\tau(\,C_1\,)\,,\mathsf{int}(\,C_2\,)\,+\tau(\,C_2\,)\,)$$

其中:函数 $\tau(C) = k/|C|$, k 是常数, |C| 是区域 C 的尺寸大小。函数 $\tau(C)$ 用来控制两个区域 C_1 、 C_2 差异性的程度, 使得区域间的差异值必须大于它们的区域内差异。

Graph-based 方法通过将图上的节点进行聚类来实现,且生成的超像素就是像素集合的最小生成树。它的运行速度很快,但不能控制超像素的数量和紧凑度。Hoiem 等人^[25]将Graph-based 用于深度估计中。

1.2 Ncut 方法

将图 G = (V, E) 分割为两个不相交的集合 $A \setminus B$, 且存在 $A \cup B = V, A \cap B = \emptyset$ 。评价分割质量好坏的最小割标准为

$$\operatorname{cut}(A,B) = \sum_{u \in A} w(u,v) \tag{4}$$

这种最小割方法是一种寻找子图间代价函数的最小划分, 但它只考虑子图间耦合度最低,而忽略了子图中节点耦合的情况,因此最小割会趋向于分离单个或者小簇顶点,如图1所示。 使用最小割方法划分图,虽然可以达到最小代价的目的,但是 违背了最优分割的标准,无法得到真正的最佳分割,于是又提出了归一化割方法。

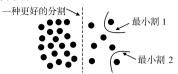


图 1 最小割不一定是最优割的情况

早期的 Neut 方法由 Shi 等人^[3]于 1997 年提出,并于 2000年作了相应的改进^[4]。Neut 对最小割算法进行改进,采用新颖的归一化割标准,同时度量了区域间的差异和区域内的相似性,很好地避免了最小割准则存在的缺陷。

Ncut 准则定义如下:

$$Ncut(A,B) = \frac{cut(A,B)}{assoc(A,V)} + \frac{cut(A,B)}{assoc(B,V)}$$
 (5)

其中: assoc(A,V)是指子集 A 中所有节点到图中所有节点的边连接权重总和, assoc(B,V) 也是类似的定义。这样定义两个区域之间的不相关性,使单个顶点分割的结果不再满足 Neut 值最小,因此不会偏向分割单个孤立的顶点。

Neut(A,B)值越低表示相似度高的顶点分配到一个子图中,相似度低的顶点分配到不同子图中,是一种较理想的评价分割质量的标准。但由于 Neut 被证明是 NP-hard 问题,且随着图中节点数目的增加,问题的求解变得异常复杂。在实际应用中,常把 Neut 准则转换为矩阵计算的形式,并求解方程的特征值和特征向量。通常选择第二个最小特征值所对应的特征向量,它代表了图划分的最优解,并取一个合适的分裂点,将特征向量离散化为两个值,根据离散化后的值将图分割为两部分。如需要再细分,可以迭代地调用该算法进行二分。

Neut 算法分割结果的特点是可以控制超像素的数量,且 形状比较规整和紧凑。但是 Neut 算法速度较慢,尤其对于尺寸比较大的图片,计算量很大。

Maji 等人^[26]在 Ncut 算法的基础上进行修改,提出了有偏差的归一化分割。相比于传统的 Ncut 算法,它存在两个优势: a)允许使用 top-down 先验信息来寻求解决方法;b)给定了未约束问题的谱方法,约束问题的解决方案可以在很短的时间内计算,且允许算法以交互的模式来运行。

Ren 等人^[1]提出了学习一个分类器来进行图像分割的方法。首先用 Neut 算法将图像映射为超像素图,它认为好的分割应从相似性、邻近性和连续性等方面来评判,因此采用了经典的 Gestalt 准则来衡量分割质量的好坏,包括区域内和区域间的纹理相似性、亮度相似性、轮廓能量和曲线连续性四方面的特征。最后,归一化这些特征,并训练相应的分类器来对分割后的结果进行分类。它是一种严格的图论分割,能约束产生许多小的、紧凑的、近似均匀区域。

1.3 Superpixel lattice 方法

对于目前的一些超像素分割算法,存在丢失原始图像重要的拓扑结构信息这一缺陷,Moorer等人^[5]提出了一种 superpixel lattice 无监督的过分割算法。该方法描述了一种能保持图像拓扑结构的贪心算法,虽然增加了拓扑信息约束条件,但是它在速度和分割精度上保持了良好的性能。

Superpixel lattice 算法输入的是图像的边界图,目的是搜寻穿过图像的最小权重路径,在边界代价图最小处分割图像。通过在垂直和水平条带两个方向搜索最优路径,不断地将图像从垂直和水平方向进行二分来得到常规网格超像素。图 2^[5]

中,(a)是将图像从左到右、从上到下依次分割图像,每条路径 将图像分为两部分,就可以得到四个区域,且在预先设定的带 条中搜索最优路径;(b)是不断在水平和垂直方向增加路径, 使图像被分割为九个区域。

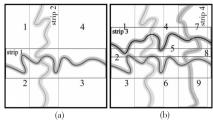


图 2 超像素网格的形成过程

在搜索最优路径的策略上, Moore 等人[5] 采用了两种方 案:s-t最小割方法和动态规划方法,前者产生任意拓扑路径,后 者则产生无回归路径。其中,最优路径需要满足三个条件:a) 每条垂直和水平路径只交叉一次;b)任意两条垂直路径不交 叉;c)任意两条水平路径不交叉。

虽然 superpixel lattice 算法已取得良好的分割效果,但其分 割质量仍然依赖于图像边界图,且隐含地规定了图像均匀分割 需要两个机制:a)图像带的分布均匀直接影响了路径的均匀分 布;b)最小代价路径策略有利于形成图像上相对较直和较短的 路径。因此, Moore 等人[27]于 2009 年在该算法的基础上加入先 验信息,提出了基于场景形状先验的超像素分割。利用概率密 度模型来描述图像物体边界的空间密度,采用一种过分割算法, 使得超像素间的密度大致相等的同时适应了局部目标边界。

随后, Moore 等人又提出了 lattice-cut^[28]方法, 它是一种无 监督的分割,采用交替选择最优策略,用单一图像割交替地在 水平或者垂直方向更新超像素边界,同时考虑了图像边界和超 像素区域内的一致性。整个超像素产生过程可以用来自图 3^[28]描述。图中,(a)首先将图像分割为均匀间距的网格超像 素,且属于同个超像素中的像素点具有相同的标签;(b)~(d) 建立马尔可夫随机场模型,不断交替地在水平和垂直方法更新 超像素晶格边界,即改变相关像素点的标签;(e)(f)对于垂直 或水平方向更新,像素的标签决定了该像素属于哪个垂直或水 平带条。Lattice-cut 方法优于现有的计算超像素网格算法,它 的性能与某些没有网格限制的分割算法具有可比性。

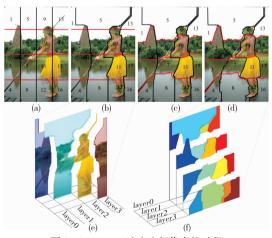


图 3 Lattice-cut 法产生超像素的过程

1.4 基于熵率的方法

Liu 等人[6]提出的基于熵率的超像素分割算法,描述了一 种基于图拓扑的能量函数。他们采用了一种新型的目标函数, 包括图像随机游走熵率和平衡项两部分。其中,熵率有利于形 成结构紧凑、均匀的集群,促使获得的超像素仅覆盖图中的单 一目标对象。平衡项促使集群有相似的尺寸,降低不平衡超像 素的个数。该目标函数可以表示为

$$\max H'(A) + \lambda B(A) \tag{6}$$

其中:A 是选择的边集 $A \in E, H'(A)$ 表示图像随机游走熵率,B(A)表示平衡项, $\lambda \ge 0$ 是一个权重值。通过最大化上述目标 函数,就可以将图像进行分割。

a) 熵率是用来衡量一个随机过程 $X = \{X, | t \in T\}$ 的不确定 性,对于离散的随机过程,熵率可以用如下公式来表示:

$$H'(X) = \lim_{t \to \infty} H(X_t | X_{t-1}, X_{t-2}, \cdots, X_1)$$
 (7)

对于图像随机游走熵率,它作为衡量集群的紧凑度和均匀 程度的标准,图 G = (V,A)上的随机游走熵率可以表示为

$$H'(X) = -\sum_{i} \mu_{i} \sum_{i} p_{i,j}(A) \log(p_{i,j}(A))$$
 (8)

其中: $\mu_i = w_i/w_T$,且 w_i 是图中与节点 i 相连的边的权重之和, w_{T} 是图中所有边权重的总和。 $w_{i,j}$ 是节点 i 到节点 j 的转移概 率,p:;是随机游走的转移概率,其定义如下:

$$p_{i,j}(A) = \begin{cases} \frac{w_{i,j}}{w_i} & \text{if } i \neq j \text{ and } e_{i,j} \in A \\ 0 & \text{if } i \neq j \text{ and } e_{i,j} \notin A \\ 1 - \frac{\sum_{j:e_{i,j} \in A} w_{i,j}}{w_i} & \text{if } i = j \end{cases}$$
 (9)

虽然添加集合 A 中的任意边能增加熵率的值,但是选择 涨幅更大的边有助于形成紧凑和均匀的集群。如图 4 所示,集 群(a)的紧凑度较高,且它的熵率值大于集群(b)的熵率值;集 群(c)的结构相对较均匀,且它的熵率值大于集群(b)的熵

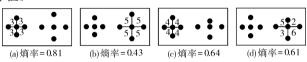


图 4 熵率对集群紧凑度和均匀性的作用

b)平衡项用于促成集群具有相同的尺寸,用如下的公式 表示:

$$B(A) \equiv H(Z_A) - N_A = -\sum_{i} pz_A(i) \log(pz_A(i)) - N_A$$
 (10)

其中: N_a 是图中连通分支的个数, Z_a 是集群成员的分布,若边 集 A 的图分割是 $S = \{S_1, S_2, \dots, S_{N_A}\}$,那么 Z_A 的分布如式 (11) 所示:

$$pz_{A}(i) = \frac{|S_{i}|}{|V|} \quad i = \{1, \cdots, N_{A}\}$$
 (11)

图 5 展示了平衡值在获得近似尺寸集群时所起的作用。 细线和粗线条连接的连通区域代表了不同的集群,(a)中集群 的平衡值大于(b)中集群的平均值,且可以看出(a)中集群分 布更均匀。

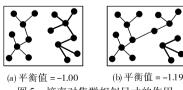


图 5 熵率对集群相似尺寸的作用

通过证明,随机熵率函数和平衡项函数都具有子模和单调 性,且引入拟阵的思想进行优化,呈现了一个有效的贪心算法。

2 基于梯度下降的超像素分割方法

对于基于梯度下降的方法,由初始的粗聚类开始,通过梯 度下降方法不断地更新聚类,直到收敛为止。

2.1 分水岭方法

分水岭方法用拓扑地形图来描述一幅图像,当应用于图像分割时,图像中每个像素的灰度值表示该点的海拔高度,每个局部极小值及其影响区域为集水盆,而集水盆的边界就形成了分水岭,大致模型可用图 6^[7] 描述。

分水岭分割算法借鉴了数学形态学的理论知识,最初由Digabel 等人^[29]将分水岭算法引入到了二值黑白图像的分析过程中;随后,Beucher 等人进行了深入的探讨,建立了较为完善的分水岭理论体系。比较经典的计算方法是 Vincent 等人^[7]提出的,采用了浸没算法的实现方案,将分水岭计算分为排序过程和浸没过程。

目前三种经典的分水岭分割算法是基于梯度的分水岭分割、基于距离变换的分水岭分割和基于标记的分水岭分割。分水岭算法的优点是简洁、复杂度低、运行时间短,且提取出的物体边缘轮廓线是封闭的,能准确定位目标物体。但是它也存在一定的缺点,分水岭分割会得到成千上万的集水盆,结果很细致,导致图像出现非常严重的过分割现象。

2.2 基于 Mean-shift 方法

Mean-shift 算法最早在 1975 年由 Fukunaga 等人 $^{[30]}$ 提出,1995 年 Cheng $^{[31]}$ 发表的文献定义了核函数和权值系数,使 Mean-shift 算法得到了广泛应用。

Comaniciu等人^[8]提出了一种无参数的、基于核密度梯度估计的快速统计迭代算法。其基本思想是在核窗口依次计算所有特征空间数据点的 Mean-shift 矢量,沿着 Mean-shift 梯度上升方向移动,直到收敛到密度最大处,如图 7 所示。通过有限次迭代计算,能够快速找到数据分布的稳定点,即模点。利用 Mean-shift 做图像分割,就是把具有相同模点的像素聚类到同一超像素中的过程。该方法在实际应用中具有较好的稳定性和抗噪性,但速度慢,且由于分割时缺少图像语义信息,分割效果不够理想,存在过分割问题。





图 7 Mean-shift 示意图

Vedaldi 等人^[9]提出 Quick-shift 分割算法类似于 Meanshift 策略的模式搜索,但是速度更快。它不断促使像素特征空间中的每一个数据点,向着能使 Parzen 密度估计增大的最近的像素移动来实现图像的分割。该算法是非迭代的,不能明确地控制超像素的大小和数量。

Wang 等人^[32]提出了各向异性核 Mean-shift 方法,而不是传统的径向对称核来估计局部密度,该方法在性能上优于传统均值漂移的方法,分割后的超像素更光滑、视觉上更符合人类感知系统。Tao 等人^[33]提出一种 Mean-shift 与 Ncut 相结合的新方法,先利用 Mean-shift 算法对输入目标图像进行预处理生成超像素,再用 Ncut 方法进行图像分割,把每个超像素看做图的节点,解决了谱聚类在图像分割时计算量大的问题,提高了算法分割效率,但分割效果并不是很理想。

2.3 Turbopixels 方法

2009 年 Levinshtein 等人^[10] 描述了一种基于几何流的水平 集方法,能快速地产生超像素。他们通过膨胀初始化种子点, 并结合曲率演化模型和背景区域的骨架化过程,将图像分割为 网格状的超像素。

Turbopixels 生成的超像素需要符合五个基本原则;a)均匀尺寸,算法应把图像分为尺寸和形状大致相同的超像素,可以通过设计几何流来膨胀初始均匀分布的种子达到此目的;b)连通性,每个超像素应该表示一个简单连通的像素集合,采用结合水平集的几何流扩张方法能确保此约束总是满足;c)紧凑性,为了使超像素最大限度地紧凑,需要使强度均匀的区域在法线向外方向产生恒定的运动;d)平滑和边缘保持,当种子增长停止时,超像素边界应该和图像边界相吻合,这需要几何流公式在边界强度弱或者没有边界的地方,曲线运动速度大,而边界强度较强的地方,曲线速度慢甚至停止,从而实现图像的分割;e)超像素不重叠,算法应该把每个像素分配到单一超像素中。因此,当两个不同的种子膨胀到即将碰撞时,应该停止边界增长。

整个算法步骤如图 8 所示,包括:

- a) 初始化等间距的种子点。
- b) 迭代以下步骤, 直至不再有进一步的演化。
- (a)第 T次演化边界;
- (b)估计未分配区域的骨架;
- (c)更新边界上像素点的速度和在边界附近未分配像素点的速度。

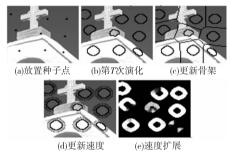


图 8 Turbopixels 算法步骤

该方法生成的超像素不仅保持了图像的局部边界,还通过一个紧凑度约束条件限制了欠分割。它的运算速度很快,算法复杂度与图像的尺寸成近似线性关系,并且对于万像素级图像,在几分钟内就可以获得高密度的超像素。

Xiang 等人^[34]提出从待分割的图像中构建多维特征图像的方法,应用于 Turbopixel 框架超像素分割中。Cigla 等人^[35]介绍了一种快速的 Turbopixel 高效图论分割方法,先采用快速Turbopixel 算法把图像分割为超像素,再建立加权图,用 Neut算法得到图的最终分割结果。

2.4 SLIC 方法

Achanta 等人^[11,12]提出了一种思想简单、实现方便的算法,将彩色图像转换为 CIELAB 颜色空间和 XY 坐标下的 5 维特征向量,然后对 5 维特征向量构造度量标准,对图像像素进行局部聚类的过程。该算法速度较快,能生成紧凑、近似均匀的超像素,具体步骤为:

- a) 初始化种子点。假设图像有 N 个像素点,预分割为 K 个相同尺寸的超像素,那么每个超像素的大小为 N/K,且每个种子点的距离近似为 $S = \sqrt{N/K}$ 。为了避免种子点处在图像的边缘位置,以及对后续的聚类过程造成干扰,需要将种子点在以它为中心的 3×3 的窗口内移动到梯度值最小的位置,同时为每个种子分配一个单独的标签。
 - b)相似度衡量。对于每个像素点,分别计算与之距离最

近的种子点之间的相似程度,将最相似种子点的标签赋给该像素。通过不断迭代该过程,直到收敛,则相似度的衡量关系如下:

$$d_{lab} = \sqrt{(l_k - l_i)^2 + (a_k - a_i)^2 + (b_k - b_i)^2}$$

$$d_{xy} = \sqrt{(x_k - x_i)^2 + (y_k - y_i)^2}$$

$$D_i = d_{lab} + \frac{m}{S} d_{xy}$$
(12)

其中: d_{lab} 为像素点间的颜色差异, d_{sy} 为像素点间的空间距离, D_i 为两个像素的相似度;S为种子点的间距,m为平衡参数,用来衡量颜色值与空间信息在相似度衡量中的比重。 D_i 取值越大,说明两个像素越相似。

为了提高算法的运算速度,对每个种子点聚类时,只在以种子点为中心的 28×28 区域内搜索相似像素点,而不是在整张图像中寻找,如图 9 所示。





(a) 在整张图像中搜索 (b) SLIC 在限定区域内搜索 图 9 减少像素的搜索范围

Ren 等人^[36]使用 GPU 和 NVIDIA CUDA 框架实现 SLIC 算法,能将速度提高 10~20 倍,促使 SLIC 能应用于实时系统。Lucchi 等人^[37]利用 SLIC 分割算法作预处理,然后以超像素为节点、空间相邻节点以边连接建立了图模型,给出相应的条件随机场定义,提出了使用核心化特征的结构图像分割。

3 实验结果对比

为了进一步了解目前常用的几种超像素分割方法的性能,本文在 Berkeley benchmark 标准数据集上进行了对比实验,验证的算法包括 entropy rate(ER)^[6]、Ncut-based(Ncut-B)^[18]、superpixel lattice(SL)^[5]、Turbopixels(TP)^[10]和 SLIC^[12]。每幅图片大小为 321 × 481,分割为约 200 个超像素的结果,如图 10 所示。可以看出,ER 算法较好地保持了图像的边界信息,但超像素的形状不太规整,分割每张图片的时间约为 1. 16 s。 Ncut-B算法很好地保持了图像边界,且随着超像素个数的增加,超像素的形状会更规则,但它对图片的处理时间较长,分割 321 × 481 大小的图片为 200 个超像素需要 2 min 左右。SL 算法的处理速度较快,大约 0. 36 s 就能将图片分割为网格状,但是它的分割效果质量严重受到输入边界图的影响。TP 和 SLIC 算法都能产生规则和紧凑的超像素,但 TP 法对边界的保持很差,

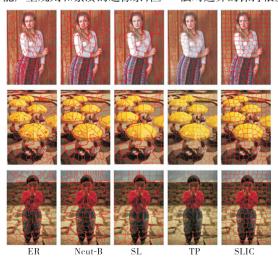


图 10 多种超像素分割算法比较

而 SLIC 从边界保持、速度和超像素形状等方面具有较高的综合评价, 比较符合人们期望的分割效果。在内存为 3.00 GB, 处理器为 Intel® Core 2 的机器上, 对上述各种超像素分割算法的时间对比结果如表 1 所示。

表1 多种超像素分割算法时间对比结果

算法	分割时间(321×481图片)/s	算法	分割时间(321×481 图片)/s
ER	1.16	TP	12.65
Ncut-B	118.20	SLIC	1.23
SL	0.36		

4 超像素的应用研究现状

超像素的概念最早在 2003 年被 Ren 等人^[1]提出,超像素算法提取了图像中感知有意义的区域,可以用来取代刚性结构的像素网格,采用超像素代替像素进行操作,可以加快现有的基于像素的算法,甚至改善一些情况的结果。近 10 年来,超像素的研究在国内外得到了迅速发展,成为了计算机视觉领域许多应用研究的关键技术。

在图像分割中,许多现有的基于图论分割算法,如Neut^[38],它的成本会随着图中节点的增加而变得昂贵,因此该算法会限制图像的尺寸大小。对于一些特定应用,如电子显微照片的线粒体分割,该类图片尺寸很大,会使基于像素网格的图像分割很棘手。在文献[14]中,采用SLIC算法将图像附近像素聚集为超像素,再将每个超像素看做图中的每个节点实现图分割,这样能有效降低图像的复杂度,使分割变得易处理,如图11 所示。在文献[15]中,利用了SLIC supervoxels 方法,能够分割数十亿个像素的大尺寸3D图像,且算法复杂性低,减少了对内存的要求,能显著提高性能。Kohli等人^[16]解决如何将属于相同标签的分割片段合并到同一目标物体的问题。









(a)原始EM图像

(b)超像素

(c)超像素图

(d)最后分割结果

图 11 超像素应用于 EM 图像分割

在人体姿态估计中, Mori 等人^[17] 先将图像分割为超像素或较大的区域块, 检测出人体轮廓并定位关节和四肢, 然后再将人体各部分组合起来。Mori^[18] 用超像素分割作预处理, 提高了在图像中模式搜索的效率和准确度, 并在静止图像中的人体姿势估计中取得了不错的成果, 如图 12^[18] 所示。



图 12 人体姿势估计示例

在目标跟踪领域, Wang 等人^[21]从中间级视觉角度出发, 提出了利用超像素来提取物体结构信息的目标跟踪算法。他 们利用一个基于超像素的外观判别模型,使跟踪器通过中间级线索来区别目标和背景。那么,跟踪的任务就转变为计算一个目标一背景信任值,并通过最大后验估计获得最好的候选结果。该跟踪算法能较好地处理目标跟踪中出现的形变、闭塞和遮挡问题,如图 13^[21]所示。Zhou 等人^[22]提出超像素驱动的水平集跟踪算法,定义一个速度函数来捕获超像素与背景或目标的相关性,该算法具有较好的鲁棒性和高效性。Liu 等人^[23]实现了在现实世界交通视频中跟踪多个车辆的情况,将语义信息引入到超像素中,有效地解决了不同车辆的频繁交叉和遮挡问题。Wang 等人^[24]通过探索基于目标周围视觉信息的超像素来解决目标跟踪问题,并提出了一个由多个部件组成的外观模型,该算法在物体形变和遮挡情况下比其他算法的效果好。



(g)transformer#17 (h)transformer#52 (i)transformer#124 图 13 超像素应用于目标跟踪

超像素还可以应用在图像处理任务的其他方面, Gu 等人^[39]将超像素运用于图像场景分类, 先把图片分割为超像素子块, 并提取图像 SIFT 特征, 形成上下文视觉特征描述符, 然后利用空间金字塔方法来表示图像并进行分类。 Tighe 等人^[40]将超像素运用在场景成分分析中, 提出了一种简单、无参数且高效的图像解析方法。 Fulkerson 等人^[20]阐述了在图像上使用超像素来定位目标和分割目标类的方法。 他们的方法在 Graz-02 和 PASCAL VOC 2007 数据集上实验的效果超过了现有的许多图像分割方法。

5 结束语

本文介绍了超像素分割算法的发展现状,着重对近年来较 具代表性的方法进行了分析。如今,超像素已成为计算机视觉 领域一项重要的工具,其广泛的应用领域和良好的实践效果受 到了国内外学者日益的关注。如何实现更精确、更快速的分割 是超像素方法研究者所面对的关键问题,算法的并行化是实现 实时处理的发展趋势。现有的不同超像素分割方法具有各自 的优势与不足,应针对特定的问题选择适当的算法以达到预期 分割效果。

参考文献:

- [1] REN Xiao-feng, MALIK J. Learning a classification model for segmentation [C]//Proc of the 9th IEEE International Conference on Computer Vision. Washington DC; IEEE Computer Society, 2003: 10-17.
- [2] FELZENSWALB P F, HUTTENLOCHER D P. Efficient graph-based image segmentation [J]. International Journal of Computer Vision, 2004, 59(2):167-181.
- [3] SHI Jian-bo, MALIK J. Normalized cuts and image segmentation

 [C]//Proc of IEEE Computer Society Conference on Computer Vi-

- sion and Pattern Recognition. Washington DC; IEEE Computer Society, 1997;731-737.
- [4] SHI Jian-bo, MALIK J. Normalized cuts and image segmentation [J].
 IEEE Trans on Pattern Analysis and Machine Intelligence,
 2000, 22(8):888-905.
- [5] MOORE A, PRINCE S, WARRELL J, et al. Superpixel lattices
 [C]//Proc of IEEE Conference on Computer Vision and Pattern Recognition. 2008:1-8.
- [6] LIU Ming-yu, TUZEL O, RAMALINGAM S, et al. Entropy rate superpixel segmentation [C]//Proc of IEEE Conference on Computer Vision and Pattern Recognition. 2011;2097-2104.
- [7] VINCENT L, SOILLE P. Watersheds in digital spaces; an efficient algorithm based on immersion simulations [J]. IEEE Trans on Pattern Analysis and Machine Intelligence, 1991, 13(6): 583-598.
- [8] COMANICIU D, MEER P. Mean shift: a robust approach toward feature space analysis[J]. IEEE Trans on Pattern Analysis and Machine Intelligence, 2002, 24(5): 603-619.
- [9] VEDALDI A, SOATTO S. Quick shift and kernel methods for mode seeking [M]//Computer Vision. Berlin: Springer-Verlag, 2008: 705-718.
- [10] LEVINSHTEIN A, STERE A, KUTULAKOS K N, et al. Turbopixels: fast superpixels using geometric flows [J]. IEEE Trans on Pattern Analysis and Machine Intelligence, 2009, 31 (12): 2290-2297.
- [11] ACHANTA R, SHAJI A, SMITH K, et al. Slic superpixels, EPFL 149300 [R]. [S. l.]: École Polytechnique Fédéral de Lausssanne (EPFL), 2010.
- [12] ACHANTA R, SHAJI A, SMITH K, et al. SLIC superpixels compared to state-of-the-art superpixel methods [J]. IEEE Trans on Pattern Analysis and Machine Intelligence, 2012, 34 (11): 2274-2282.
- [13] LEVINSHTEIN A, SMINCHISESCH C, DICKINSON S. Multiscale symmetric part detection and grouping[J]. International Journal of Computer Vision, 2013,104(2):117-134.
- [14] LUCCHI A, SMITH K, ACHANTA R, et al. A fully automated approach to segmentation of irregularly shaped cellular structures in EM images [C]//Medical Image Computing and Computer-Assisted Intervention. Berlin; Springer-Verlag, 2010; 463-471.
- [15] LUCCHI A, SMITH K, ACHANTA R, et al. Supervoxel-based segmentation of mitochondria in EM image stacks with learned shape features[J]. IEEE Trans on Medical Imaging, 2012, 31(2): 474-486
- [16] KOHLI P, TORR P H S. Robust higher order potentials for enforcing label consistency [J]. International Journal of Computer Vision, 2009, 82(3): 302-324.
- [17] MORI G, REN Xiao-feng, EFROS A A, et al. Recovering human body configurations: Combining segmentation and recognition [C]// Proc of IEEE Computer Society Conference on Computer Vision and Pattern Recognition. Washington DC: IEEE Computer Society, 2004: 326-333.
- [18] MORI G. Guiding model search using segmentation [C]//Proc of the 10th IEEE International Conference on Computer Vision. Washington DC; IEEE Computer Society, 2005;1417-1423.
- [19] PANTOFARU C, SCHMID C, HEBERT M. Object recognition by integrating multiple image segmentations [C]//Proc of the 10th European Conference on Computer Vision. Berlin: Springer-Verlag, 2008: 481-494.
- [20] FULKERSON B, VEDALDI A, SOATTO S. Class segmentation and

- object localization with superpixel neighborhoods [C]//Proc of the 12th International Conference on Computer Vision. Washington DC: IEEE Computer Society, 2009: 670-677.
- [21] WANG Shu, LU Hu-chuan, YANG Fan, et al. Superpixel tracking [C]//Proc of IEEE International Conference on Computer Vision. 2011;1323-1330.
- [22] ZHOU Xue, LI Xi, CHIN T J, et al. Superpixel-driven level set tracking [C]//Proc of the 19th IEEE International Conference on Image Processing. 2012;409-412.
- [23] LIU Li-wei, XING Jun-liang, AI Hai-zhou, et al. Semantic superpixel based vehicle tracking [C]//Proc of the 21st International Confe-rence on Pattern Recognition. Berlin: Springer-Verlag, 2012:2222-2225
- [24] WANG Wei-jun, NEVATIA R. Robust object tracking using constellation model with superpixel [C]//Proc of the 11th Asian Conference on Computer Vision. Berlin; Springer-Verlag, 2012;191-204.
- [25] HOIEM D, EFROS A A, HEBERT M. Automatic photo pop-up[J]. ACM Trans on Graphics, 2005, 24(3):577-584.
- [26] MAJI S, VISHNOI N K, MALIK J. Biased normalized cuts [C]// Proc of IEEE Conference on Computer Vision and Pattern Recognition. Washington DC: IEEE Computer Society, 2011:2057-2064.
- [27] MOORE A, PRINCE S. Scene shape priors for superpixel segmentation [C]//Proc of the 12th IEEE International Conference on Computer Vision. Washington DC; IEEE Computer Society, 2009;771-778.
- [28] MOORE A, PRINCE S J D, WARRELL J. Lattice cut -constructing superpixels using layer constraints [C]//Proc of IEEE Conference on Computer Vision and Pattern Recognition. 2010:2117-2124.
- [29] DIGABEL H, LANTUEJOUL C. Iterative algorithms [C]//Proc of the 2nd European Symposium on Quantitative Analysis of Microstructures in Material Science, Biology and Medicine. Sturrgart: Riederer-Verlag, 1978.
- [30] FUKUNAGA K, HOSTETLER L. The estimation of the gradient of a density function, with applications in pattern recognition [J]. IEEE Trans on Information Theory, 1975, 21(1): 32-40.

(上接第5页)

- [14] DESNOS A, FILIOL E, LEFOU I. Detection of an HVM rootkit (aka BluePilllike) [J]. Computer Virology, 2011, 7(1):23-49.
- [15] KYTE I, ZAVARSKY P, LINDSKOG D. Enhanced side-channel analysis method to detect hardware virtualization based rootkits [C]// Proc of World Congress on Internet Security. 2012;192-201.
- [16] RUTKOWSKA J. Red Pill or how to detect VMM using (almost) one CPU instruction [EB/OL]. (2004-11-17). http://www.securiteam.com/securityreviews/6ZooH20BQS.html.
- [17] RUTKOWSKA J. Blue Pill detection [EB/OL]. (2006-03-12). http://theinvisiblethings. blogspot. com/2006/08/blue-pill-detection.
- [18] QUIST D, SMITH V. Detecting the presence of virtual machines using the local data table [EB/OL]. http://index-of.es/Misc/vm.pdf.
- [19] FERRIE P. Attacks on virtual machine emulators [EB/OL]. http:// www. symantec. com/avcenter/reference/Virtual_Machine_Threats. pdf.
- [20] ADAMS K. BluePill detection in two easy steps [EB/OL]. (2007-07-02). http://x86vmm. blogspot. com/2007/07/bluepill-detection-in-two-easy-steps%. html.
- [21] Virtualization technology for AMD architecture [EB/OL]. http://download.microsoft.com/download/9/8/f/98f3fe47-dfc3-4e74-92a3-088782200fe7/twar05014_winhec05.pdf.

- [31] CHENG Yi-zong. Mean shift, mode seeking, and clustering [J]. IEEE Trans on Pattern Analysis and Machine Intelligence, 1995, 17(8): 790-799.
- [32] WANG Jue, THIESSON B, XU Ying-qing, et al. Image and video segmentation by anisotropic kernel mean shift [C]//Proc of European Conference on Computer Vision Berlin: Springer-Verlag, 2004:238-249.
- [33] TAO Wen-bing, JIN Hai, ZHANG Yi-ming. Color image segmentation based on mean shift and normalized cuts[J]. IEEE Trans on Systems, Man, and Cybernetics, Part B: Cybernetics, 2007, 37(5):1382-1389.
- [34] XIANG Shi-ming, PAN Chun-hong, NIE Fei-ping, *et al.* Turbopixel segmentation using Eigen-images [J]. IEEE Trans on Image Processing, 2010, 19(11); 3024-3034.
- [35] CIGLA C, ALATAN A A. Efficient graph-based image segmentation via speeded-up turbo pixels [C]//Proc of the 17th IEEE International Conference on Image Processing. 2010;3013-3016.
- [36] REN C Y, REID I. gSLIC: a real-time implementation of SLIC superpixel segmentation [R]. Qxford: Department of Engineering, University of Oxford, 2011.
- [37] LUCCHI A, LI Yun-peng, SMITH K, et al. Structured image segmentation using kernelized features [C]//Proc of the 12th European International Conference on Computer Vision. Berlin: Springer-Verlag, 2012: 400-413.
- [38] BOYKOV Y Y, JOLLY M P. Interactive graph cuts for optimal boundary and region segmentation of objects in N-D images [C]// Proc of the 8th IEEE International Conference on Computer Vision. 2001: 105-112.
- [39] GU Guang-hua, LI Feng-cai, ZHAO Y, et al. Scene classification based on spatial pyramid representation by superpixel lattices and contextual visual features [J]. Optical Engineering, 2012, 51(1): 017201-1-017201-8.
- [40] TIGHE J, LAZEBNIK S. Superparsing: scalable nonparametric image parsing with superpixels [C]//Proc of the 11th European Computer Vision. Berlin: Springer-Verlag, 2010: 352-365.
- [22] MYERS M, YOUNDT S. An introduction to hardware-assisted virtual machine (HVM) rootkits[R]. [S.l.]; WhitePapers DB. 2007.
- [23] LI He-shuai, ZHU Jun-hu, ZHOU Tian-yang, et al. A new mechanism for preventing HVM-aware malware [C]//Proc of the 3rd International Conference on Communication Software and Networks. 2011: 163-167.
- [24] SHEN J, LIPASTI M. Modern processor design: fundamentals of superscalar processors [M]. [S. l.]; McGraw-Hill, 2005.
- [25] MILENKOVIC M, MILENKOVIC A, KULICK J. Demystifying Intel branch predictors [EB/OL]. http://www.ece.wisc.edu/~wddd/2002/final/milenkovic.pdf.
- [26] FERRIE P. Attacks on more virtual machine emulators [EB/OL]. (2007-08-31). http://pferrie. tripod. com/papers/attacks2. pdf.
- [27] BULYGIN Y, SAMYDE D. Chipset based approach to detect virtualization malware [R]. [S.1.]: Intel Corporation, 2008.
- [28] GARFINKEL T, PFAFF B, CHOW J, et al. Terra; a virtual machine-based platform for trusted computing [C]//Proc of the 19th ACM Symposium on Operating Systems Principles. New York; ACM Press, 2003;193-206.
- [29] 于森,戚正伟. NewBluePill 深入理解硬件虚拟机[M]. 北京:清华 大学出版社,2011:55.
- [30] WMware. Timekeeping in VMware virtual machines [EB/OL]. http://www.vmware.com/pdf/vmwaretimekeeping.pdf.