摘　要

本文的研究工作主要针对的是图像目标识别，希望能提高目标识别的精度与鲁棒性。研究的主要内容为：假定在一幅图片中存在某个既定目标，则要求算法识别目标位置并尽可能细致地描绘目标的轮廓。所需的知识范围涉及图像处理、机器学习、信号滤波、随机过程与统计学等、模式识别等多方面。国内外都在积极做该方向研究，但一直存在模型鲁棒性差，而且大部分基于像素级别的研究。本文将通过学习的方法训练一个用于目标识别的模型，模型中是考虑了许多种可能的数据特征，包括 DenseSIFT、DenseColor 和 Textons，以及它们的任意组合方式。另外，本文不是基于像素级别，文中提供了多种提取超像素区域的方法，包括 SLIC 与 QuickShift，是基于超像素级别的研究。文中使用的训练模型有支持向量机和条件随机场。本文研究方法为：理论、实践、改进、理论的循环迭代模式。实验以单目标识别问题为主，多处采用模块化的小实验来证各部分的理论，最终整合形成工程，实验设计以方便选择合适参数为重心。通过最终的整合工程，预期能以一个较高的平衡精度实现两类的目标识别问题。最终，将尝试着训练一个多类目标的识别模型，用于多类目标识别。

关键词：目标识别；超像素；支持向量机；条件随机场

Image Object Recognition - Extract Features For Superpixel

Abstract

In this paper, research work is focused on image object recognition, hoping to improve the accuracy and robustness of object recognition. The main contents are: assume the existence of an image in a stated objective, which requires algorithms to identify possible location and contours of the object depicted in detail. Required knowledge covering image processing, machine learning, signal filtering, stochastic processes, and statistics, etc., pattern recognition and other aspects. At home and abroad, a lot of active research in that direction, but there has been a problem that robustness is poor and largely based on pixel-level research. This paper will train a model for object recognition, the model considered a number of possible samples' features, including DenseSIFT, DenseColor and Textons, and possible combinations of any of them. Additionally, this article is not based on the pixel level, the paper provides a variety of methods to extract superpixel regions, including SLIC with QuickShift. It is based on the superpixel leveled research. The training method we used in this paper including SVM and CRF. The research method here is: theory, practice, improvement, theory and iterated. Experiments with a single object recognition based, multiple modular small experiment to prove the theory of the various parts, and ultimately the formation of integrated engineering, experimental design to facilitate select the appropriate parameters as the focus. By the end of the paper, we are expected to be able to achieve a high balance accuracy on double-class object recognition. Ultimately, we will try to train a multi-class object recognition model for multi-class object recognition.

Key Words：Object Recognition; Superpixel; SVM; CRF

目　　录

插图或附表清单

引　言

本文搭建一个使用训练的方式实现的图像目标识别框架，主要为了提高两类图像目标识别的数据可扩展性和精度。涉及内容包括特征提取、图像分割及标签分配、聚类、训练等多过程。之前的研究大部分国内外研究都在使用一种或较少的特征类，相比较而言，本文框架提供了任意多种特征的组合接口。Koen E. A.[1]等人虽研究使用了多种特征用于识别，但集中在讨论颜色特征上，不够全面。Fulkerson 等人[2]的工作中中只使用了简单的灰度域的直方图特征用于训练。另一方法，之前的工作几乎都局限在像素级的图像识别上，09 年之后有所改观，本文的工作与 Fulkerson 的工作类似，在基于区域匹配的超像素级别上实现训练识别过程，Fulkerson 文中只提供了 QuickShift 的超像素分割方法，本文提供了其它的比如 SLIC 的超像素分割方法，这两种方法在不同的数据场景下分割效果截然不同，因此针对具体的数据集可使用不同的超像素方法，而且本文提供新的超像素提取方法接口。因为有“具体问题具体分析”，针对不同的应用领域，图像识别方法纷繁陈杂，很少提供数据可扩展性的框架。本文欲通过对一些特征提取方法、超像素分割方法的整合，形成扩展性的描述超像素的统计直方图特征，使用支持向量机和条件随机场模型在超像素基础上训练。通过框架搭建，结合主观与客观结合的实验参数选择，最终从主观上选择一组合适的参数对几个较大的数据集训练测试，预期这几类数据集能在超像素级别基础上都能达到一个较好的识别精度。通过本文的研究，能体会当前前言图像识别领域的一些优秀成果，本文正是在这基础上进行改进与融合的一个过程，最终将在 graz-02 数据集上和 sowerby 数据集上测试和验证本文结果。

1 绪论

1.1 图像目标识别及国内外现状

目标识别是图像处理领域比较新且难的一大课题，没有坚实的专门的理论基础支撑，现仍然集中在尝试与探索阶段。本文中研究的目标识别指确定图像中目标在图像中的位置并能尽可能细致地描绘出目标轮廓。目标识别与目标跟踪有所不同，目标识别不仅需要识别目标的位置，还要描出目标的轮廓，而目标跟踪的核心是确定目标位置的移动。它也与图像分割不同，分割只需要完成具有实际差异性区域的边缘界定工作，而目标识别除了界定区域之外还需要甄别分割区域所属类型是目标还是背景。目标识别包括目标跟踪的目标位置检测过程，也包括图像分割中的目标边界检测过程，是两者的一种融合。目标识的过程别与人类辨别各种不同物体过程类似，使用计算机智能地实现图像识别也同样经历这样的过程——信息获取、信息加工、特征提取和比较判别的过程。

图像识别的历史比较久，大约在 20 世纪 50 年代左右，人们便已经开始投入到二维图像的识别研究过程当中，当时工作主要集中在识别工件表面缺陷的等，大部分集中在急切存在需求工业领域，而且识别的目标也相对简单。但随着图片技术的发展，图像背景越来越复杂，图像识别变得越来越困难，现在的大部分图像识别的方法都是集中借鉴其它领域的理论，譬如概率统计、自然语言处理、生物仿真，其本身并未形成专业性的理论。但随着机器学习技术的发展和推动，图像识别领域引入了更多新的思想和方法，针对的特定应用领域也研究更加广泛，智能机器人识别、遥感识别等领域都需要基本的图像识别技术的发展做支撑。

最初应用在图像识别中的最简单的方法是数学拟合，而后新判别方法的提出对图像识别领域是重要突破。支持向量机[3]是一种很好的线性判别方法，支持向量机中的核函数引入更为提高中小数据集的分类精度提供可能。[4]提供了一种基于先验特征匹配的目标识别方法。近几年，马尔科夫模型与条件随机场[5][6]模型也被引入到图像识别的研究当中，实践证明能获得很好的效果。另一方面，08 年之前大部分研究成果都集中在像素级，随着词袋模型从自然语言处理中引入，使得条件随机场能方便地构建在区域图结构之上，实现超像素级的图像识别，同时将聚类技术用到图像识别中的例子也屡见不

鲜[7]。最近一两年，一些结构化的方法[8]也开始在图像目标识别中开始流行，不过经常用于辅助求解支持向量机或条件随机场的问题。虽然图像识别的技术一直在提升，但直到现在，该领域仍然存在纵多棘手的问题，比如复杂图像识别精度低、算法训练时间过长，数据鲁棒性差等。国外的伯克利大学、牛津大学等都在积极做相关方面的研究，国内中国科学院、北京大学等高校研究机构正致力于特定领域的图像识别研究，比如太空遥感图像识别等。图像识别的现状用一句化描述为：尚处在尝试求索阶段，悬而未决的问题仍多于已解决的问题。

1.2 图像目标分类及关键技术

当前已经提到许多目标识别的方法。从构造特征的角度，目标识别方法可简单地分为两种：基于像素的识别和基于区域的识别。基于像素的识别方法主要是如何提取和构造像素的特征，使用比如灰度、颜色、SIFT 特征以及纹理特征等。基于区域的识别方法，这种方法需要经历两个步骤，第一是采用一种有效的图像分割算法对图像分割成多个区域，第二是采用一种区域特征对区域进行描述。从另一个角度，图像识别也可以划分为基于决策理论的方法和基于结构的识别方法。决策理论的方法使用譬如模板匹配分类器、贝叶斯分类器、神经网络分类器和支持向量机等分类机制，需要训练数据建立分类模型，再使用分类模型对新入图像进行分类判别。而基于结构的识别方法则使用诸如串结构或图结构对图像特征进行描述。

在 20 世纪 50 年代时，人们早将数学拟合技术用与图像识别。从那之后，模板匹配成为图像识别中最主要的技术，模板匹配模型需要通过经验设定目标模板，然后通过相似度测量判断实际目标与模板的匹配程度，这是模式识别中一种最原始的的技术。到如今，模板匹配可能只体现在图像目标识别技术中的一个小的部分，许多融合其它领域的技术推动着图像识别的发展。数理统计与随机过程是图像识别中不可或缺的技术，比如直方图、后验概率最大、马尔可夫性等概念都已经在图像识别当中使用。图像本身也是一种信号，信号处理尤其是自然语言处理领域和通信信号处理中率先使用的技术，譬如滤波器、最大熵模型等，都在被图像识别领域使用，本文中将论及的 Textons 特征就是通过滤波器卷积获得的。当然，图像识别还与图像分割密不可分。最近 10 年，大量的基于机器学习的方法被用到图像识别的研究当中，譬如支持向量机、均值聚类算法以及

神经网络等。综合来说，图像的目标识别涉及图像分割、图像特征提取、数理统计、信号处理、模式识别[9]、机器学习、组合优化等纵多关键技术，而且生物智能等跨度较大的技术也渐渐在图像识别中使用。

尽管有这许多可用于图像目标识别的技术，但在研究图像目标识别的问题时，我们仍然没有办法做到“一手抓一大把”，每种图像目标识别的技术或方法都是适用在特定的数据集、特定的领域当中，不同的领域中的图像是千差万别的。每个领域都有与之特定的有效的图像识别技术，当我们把图像目标识别的领域确定后，图像目标识别问题将变得愈加简单，比如人脸识别与车牌识别，都是集中在一个很小的特定的应用当中，无疑这些也是目前图像目标识别中做得比较好的。

1.3 课题工作与本文组织

本文在吸收国内外前人工作经验的基础上，搭建了一个综合性的、可多特征融合的、多种超像素可选的、数据可扩展性较强的图像目标识别平台，目的在于提出一种以训练为基础的自学习的图像目标识别框架，并希望提高数据的可扩展性，和一些特定数据集的识别精度。如前文所述，目标识别的特定应用领域确定后，目标识别将变得稍微简单，提供统一的识别方法几乎是不可能的，本文并不是为了提供一个万全的对任意数据可行的办法，而是提供一种相对而言较稳定的办法，通过数据多特征融合、超像素可选的方式为具体的识别问题提供更大的自由空间。本文在原理或技术的选择上不约束于特定的数据集及其应用领域，但这种不确定性也无疑增加了目标识别的难度。

本文正文共分为 6 部分。

第一部分：绪论。简要介绍图像识别的概念，历史，国内外发展现状及本文主要研究的工作和目的。

第二部分：总体框架模型。综述本论文用于研究搭建的设计框架，简述本文将使用的方法与技术及其相关的概念。

第三部分：超像素特征提取。介绍提取本文中使用到的超像素特征的方法及流程，并给出一些中间的实验效果。

第四部分：支持向量机。本文使用支持向量机对提取的特征分类，该部分将详述支持向量机分类的细节，包括对交叉验证方法的描述，以及如何在本文中使用。

第五部分：条件随机场。论述条件随机场的原理，描述如何在本文的框架中使用条件随机场对图像识别的结果做进一步优化。

第六部分：实验。对比单种特征及多种不同的特征组合时图像识别的效果，并给出一些实验表格和图像识别结果。

2 总体框架模型

本文采用基于区域与决策理论的图像目标识别方案，在不考虑本文使用特定数据集的情况下，该方案可用于大部分的两类及多类目标识别问题的数据集当中。方案中灵活提供可融合的多种特征提取方法、可选的超像素提取方法、聚类方法和判决模型。在该方案中，数据分为三类：训练数据，测试数据，新输入数据。考虑给定的数据集（本文中将以 graz-02 数据集为基准，但也可用于其它数据集）以及基准事实（groud truth），将数据集中的一部分划分为训练数据，另一部分则为测试数据，训练数据作为一种已知事实用于训练支持向量机模型与条件随机场模型，其结果是获得可预测的模型，测试数据用来评估该模型的分类识别效果，作为衡量模型好坏的一个准则。基准事实是作为衡量目标识别准确度的样本，是训练的基础依据也是目标识别的目的，本文使用的基准事实也称为掩码图片（mask image）。对于新输入图像，不可能确定基准事实，因为这正是图像目标识别要做的工作，为此新输入图像将通过已训练出的支持向量机模型或条件随机场模型进行预测，预测的结果就是类似于训练数据集中的基准事实一样的东西，从而达到图像目标识别的目的。

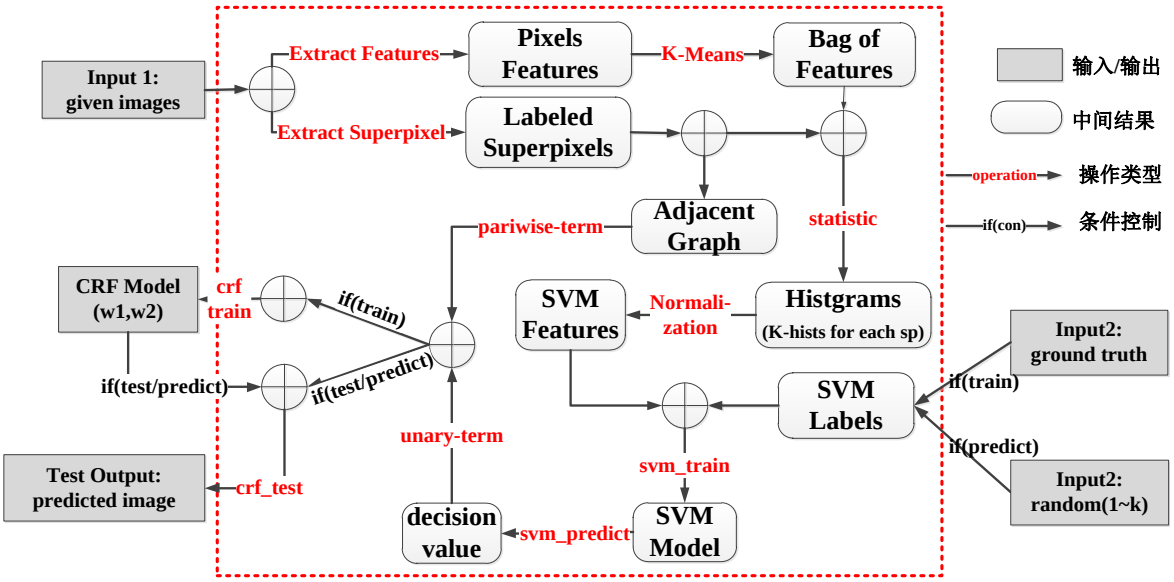

图 2.1 图像目标识别的总体设计框架

首先，样本数据都使用相同的参数提取像素特征（Pixels Feature），标定的超像素（Labeled Superpixels）是通过图像分割获得的。像素特征是点特征，可使用尺度不变转换特征（Scale-invariant feature transform, SIFT）或颜色特征等，结果每个提取特征的像素点特征使用一个向量描述，称为特征描述子。标定的超像素是图像标定的区域的集合，换句话说，为每个分割区域分配唯一的标签就是对超像素的标定过程，超像素图的尺寸与原图像相同，根据超像素图可以构造图的数据结构，用于后面的条件随机场模型当中。

第二，使用聚类算法对特征点聚类，形成特征包（Bag of Features）。K-Means 聚类算法是一种无监督的聚类学习方法，本文为提升运算速度将使用其改进版本。通过特征点的坐标信息与特征聚类的类别信息可以将超像素与特征包联系起来，对于每个超像素，都只对所有分布在超像素内的特征点统计直方图（Histogram），统计过程以特征包的聚类为基础，因此直方图的维度就为特征包的聚类数目，纵坐标为统计结果，描述所属类中有多少特征点在统计的超像素的物理坐标范围内。直方图就作为描述超像素的区域特征，经过这种转化，像素点特征被转化为超像素特征，将像素点特征转化为超像素特征的相对于像素特征的好处是：对于图像这种大的数据，如果使用像素点特征直接去训练，时间复杂度是很难想象的。另一方面，为了使用条件随机场的需要，标定的超像素图使用超像素邻域的概念建模形成邻接图（Adjacent Graph），邻接图描述了不同超像素之间的是否相邻的关系，若超像素之间相邻，通过邻接图的权重能描述超像素之间的相似度。

第三，支持向量机（Support Vector Machine, SVM）的训练过程。训练数据要求提供基准事实，比如本文中将使用 Graz-02 的基准事实也是来自于 Graz-02 的中提供的掩码图片（mask image），掩码图片指已经将原图像中的不同类别使用不同颜色、相同类别使用相同颜色划分的图片。我们可以通过使用掩码图片获得每个超像素的标签，以此作为支持向量机训练用标签（SVM Labels）。图像目标识别的目的也就是对新输入的图片标定获得掩码图片的过程，因此新输入的图片掩码图片未知，因此在支持向量机的标签中可使用随机值，这对结果没有影响。另一方面对于数据的处理，使用支持向量机前

有必要对直方图特征数据归一化。当然，在支持向量机训练过程中还要一系列需要考虑的问题，譬如参数的选择与优化，评估函数的选择等。

第四，使用支持向量机的训练结果模型已经能获得很好的效果了，添加条件随机场（Conditional Random Field, CRF）是为了优化图像目标的边缘。前面已经提到将超像素转化为邻接图，条件随机场模型就构建在邻接图基础上。条件随机场优化过程是通过计算能量函数的最小值，而能量函数则是由一元势项（unary-term）与二元势项（pairwise-term）两项构成的和。邻接图的权值提供条件随机场所需的二元势项，一元势项通过支持向量机的输出判决函数值得到。

如图 2.1即为本文的总体设计框图，虚线框内描述的是流程，框外描述的是输入与输出。给定数据集一边提取像素点特征、聚类，另一边提取超像素，两者结合统计形成直方图，归一化的直方图用于支持向量机训练，用于训练和测试图像的标签通过掩码获得，新输入图片的标签则取随机值。经过支持向量机的训练后获得条件随机场所需的一元势项，二元势项通过超像素的邻域关系获得。在训练时，一元势项与二元势项给定后可用于训练条件随机场的参数，在测试或预测中，条件随机场模型直接给出预测结果。

3 超像素特征提取

3.1 提取像素点特征

考虑到实际的需要，本设计现在已提供 3 种像素点特征，根据实际数据需求可选择使用一种或多种的组合。有一种特征是对图像旋转、尺度伸缩具有强不变性，叫尺度不变转换特征（SIFT）[10]，本文使用它的一种变化版本，DenseSIFT，两者在提取描述特征的向量的算法上没有任何差别。DenseSIFT 特征本身不具有任何的颜色信息，有两种方式可以将颜色信息考虑进系统中，第一，使用 RBG-DenseSIFT，对原始图像在 RGB 空间中每维单独采样，但这使得特征向量的维度扩大为原来的 3 倍，第二，使用 DenseColor 单独提取颜色特征，本文中主要使用到 DenseColor。本文还提供第三种特征——Textons[11]，与前两种特征不同，这是一种纹理特征，考虑从频域提取特征信息。为方便处理，我们都使用两个量来对三种特征做统一性描述，使用帧（frame）表示每个特征的一些基本信息，比如特征点坐标，特征点方向（如果有的话），特征点的参数信息等，使用特征描述子（discriptors）表示每个特征的特征向量表述，这是像素点特征数字化的描述，也是在特征提取过程中我们关注的重点。

3.1.1 DenseSIFT 特征

SIFT 特征是一种坚持局部特征的算法，通过在尺度空间寻找关键点（keypoints），提取关键点的位置、尺度与方向 3 种信息，对关键点的周围区域统计直方图获得特征的向量信息，向量信息称描述子。SIFT 有许多变化版本[12]，DenseSIFT 特征避免了 SIFT 特征寻找关键点的操作，对图像进行高密度的等间隔采样，使用与 SIFT 相同的方法提取尺度、方向和特征的向量信息，DenseSIFT 的采样间隔的选择需要仔细考量。

DenseSIFT 特征的尺度与尺度空间有很大关系，尺度空间指使用高斯函数与原始坐标空间的卷积运算（LOG），如下式，

$$L(x,y,\delta) = G(x,y,\delta) * I(x,\delta) \tag{3.1}$$

高斯函数中的δ取不同的值，卷积后的值则不同，从而形成不同尺度的尺度空间。更常用的尺度空间是高斯差分（DOG）尺度空间，DOG 尺度空间是 LOG 尺度空间的近似，改用高斯差分对原始坐标进行卷积运算，选定一个尺度δ，使用kδ尺度的高斯函数与δ的高斯函数作差分运算，差分运算结果与原始坐标空间卷积，如下式所示，

$$D(x,y,\delta)=(G(x,y,k\delta)-G(x,y,\delta))*I(x,y)=L(x,y,k\delta)-L(x,y,\delta) \tag{3.2}$$

本文中使用的 DenseSIFT 使用的是 DOG 尺度空间，尺度空间的δ 值就描述了 DenseSIFT 所需的尺度信息，尺度信息保证了图像的伸缩不变性。

DenseSIFT 特征的方向信息提取方法：以关键点为中心，用直方图统计关键点周围区域特征点的梯度方向。梯度方向的范围是 0～360 度，假设每 45 度一个方向，则直方图中总共 8 个柱，统计关键点周围分别属于 8 个方向的关键点的点数。使用直方图的峰值则代表关键点的主方向。关键点的方向信息保证了图像的旋转不变性。

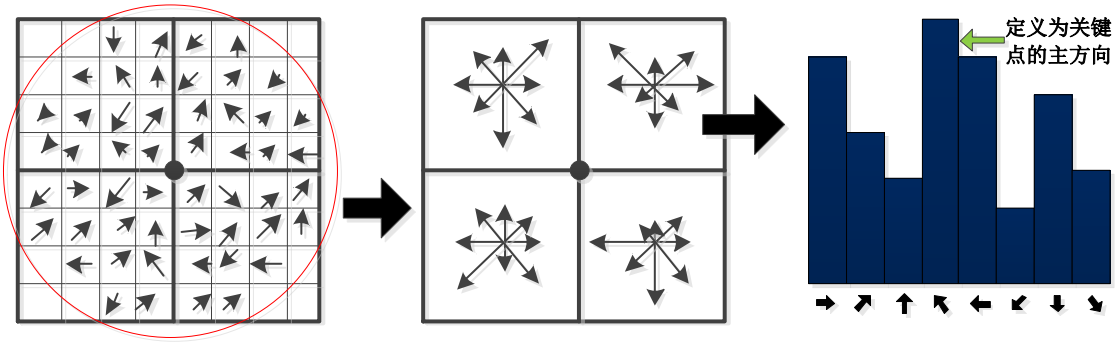

图 3.1 SIFT 或 DenseSIFT 特征描述子的形成过程

DenseSIFT 特征的描述子是用于图像识别中的关键特征，DenseSIFT 描述子的生成过程为：（i）将坐标轴旋转为关键点主方向，确保旋转不变性，旋转变换的表达式为

$$\begin{pmatrix} x' \\ y' \end{pmatrix}=\begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & -\cos\theta \end{pmatrix}\times\begin{pmatrix} x \\ y \end{pmatrix} \tag{3.3}$$

θ表示特征主方向。（ii）计算对于每个关键点产生 128 个数值数据，形成 128 维的 DenseSIFT 特征描述子。在关键点点周围划取 16x16 大小的窗口，以 4x4 大小的小窗口统计得到一个 8 方向的梯度图，共 16 个小窗口，因此关键点特征的维度为 8x16=128。

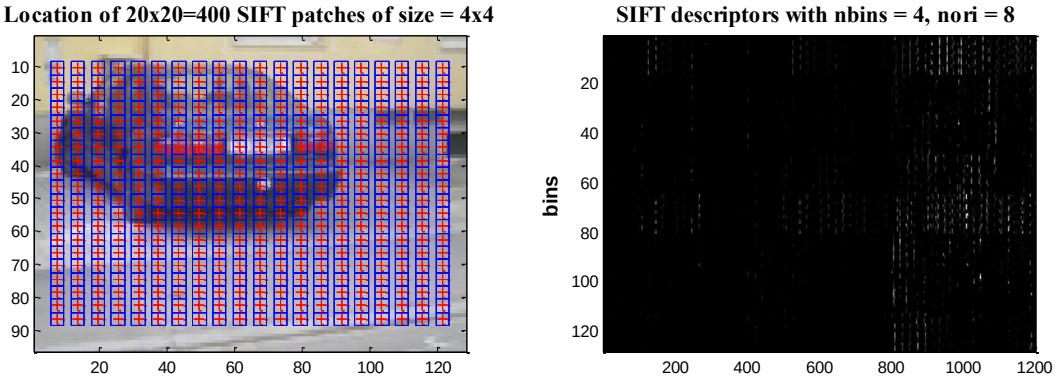

图 3.2 DenseSFIT 特征. 左图为 DenseSIFT 特征位置，右图为 DenseSIFT 特征描述符.

DenseSIFT 特征在方向与尺度变换上具有不变性，但不具有任何的对图像颜色信息的描述，有一种方法，使用 RGB-DenseSFIT，即在 RGB 空间中将 R、G、B 每一个维度单独使用 DenseSIFT，然后将 DenseSIFT 特征拼接起来形成 RGB-DenseSIFT 特征。但是，RGB-DenseSIFT 使特征的维度变为原来的 3 倍（384 维），因此本文将单独将描述颜色的 DenseColor 特征纳入到框架当中。

3.1.2 DenseColor 特征

与 DenseSIFT 一样，DenseColor 特征也是高密度地对原图像采样，对每个特征点选定一定大小的窗口区域，将大窗口划分为多个子窗口，每个子窗口统计 N 维的直方图。不同的是，DenseColor 在选定区域内统计的是颜色灰度的直方图而不是梯度方向图，因此产生的是颜色不变信息。因为不需要变化的尺度信息，DenseColor 在算法前部分使用 LOG 而非 DOG 计算尺度空间。DenseColor 特征的描述子维度是可选择的，这取决于对颜色量化的精度，通过对颜色量化，量化后为同一个值的灰度范围被统计到同一直方图柱面上。

对于灰度图像，对每个特征点，只需要在一个固定的周围区域内统计灰度值；对于彩色图像，可以有多重彩色空间用来描述，比如 RGB、LUV 等。许多产生图像的自然环境会影响到图像的色彩，比如光照强度、拍摄角度等，从而影响到图像的识别效果[1]。为此在必要的时候使用对光照强度不变、图像旋转不变、颜色平移不敏感的颜色空间。RGB 颜色空间的不具有上述的不变性，只适合在图像拍摄环境变化不大的场合。LUV 彩色空间使用亮度与色度描述，颜色平移变化不敏感，但对强度变化敏感。

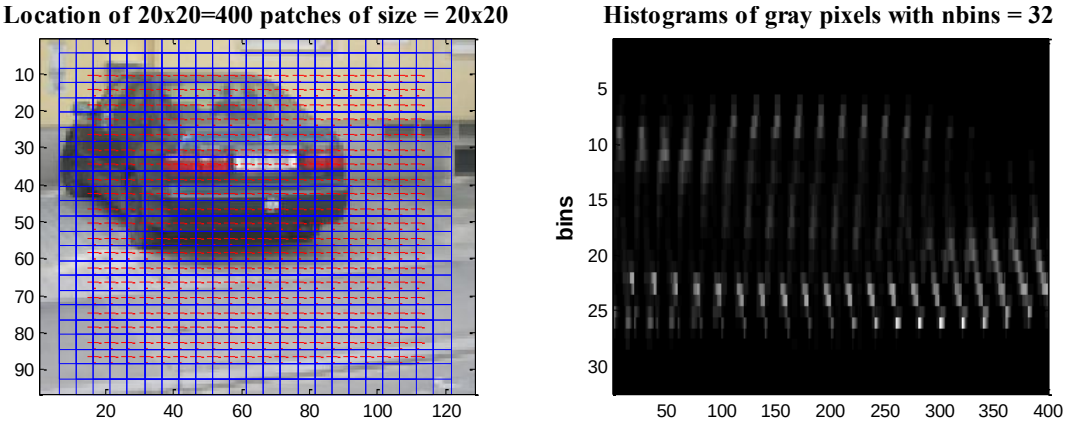

图 3.3 DenseColor 特征. 左图为 DenseColor 特征的位置，右图为 DenseColor 特征描述符.

3.1.3 Textons 特征

DenseSIFT 特征和 DenseColor 特征都是基于空域信息，与它们不同的，Textons 特征使用的是频域信息。Textons 使用了一组滤波器，训练图片通过滤波器组，与滤波器发生卷积运算，假设滤波器组共有 n 个滤波器，则会产生 n 个响应，假设图像尺寸为 MxN，则一幅经过滤波器组后的响应维度为 MxNxn。对于每个特征点，在每个响应中对应的物理坐标点上取一个值组成 Textons 特征，若滤波器响应后的维度为 MxNxn，则 Textons 特征描述子的维度为 n。

本文使用的滤波器组为 RFS（Root Filter Set），RFS 滤波器组共有 38 个不同类型、尺度或方向的滤波器。详细的，RFS 滤波器组中包括：$\delta=10$ 高斯滤波器和 LOG 滤波器各一个，这将在与原图像卷积后产生 2 个滤波器响应，高斯滤波器与 LOG 滤波器都是四周对称的滤波器（图 3.4），因此无任何的纹理上的方向特征信息；还有一个在 3 个尺度上 $(\delta_x,\delta_y)=\{(1,3),(2,6),(4,12)\}$ 的边缘滤波器以及一个同样的 3 个尺度上的条形滤波器，这两个滤波器都对每个特征点位置的 6 个方向进行卷积运算，而且这些滤波器都是中心对称类型，融合了纹理的方向性特征，共将产生 2x3x6=36 共个滤波器响应。将上述滤波器的响应个数加起来则构成了图像的 38 维的特征向量，对滤波器响应的按像素位置高密度等间隔采样就构成 Textons 特征的描述符，当然采样时也能确定特征点的坐标信息用于填充 frame。因此使用 RFS 滤波器的 Textons 特征维度为固定的 38。除 RFS 滤波器外，还有其它许多可用的用于产生 Textons 纹理特征的滤波器。

Textons 特征从频域的角度对图像特征点进行描述，对于纹理特征较弱的图片，如果单独使用 Textons 的效果将会比较差，因此，在对图像信息不清晰的情况下，往往将 Textons 特征与其它特征结合起来使用。

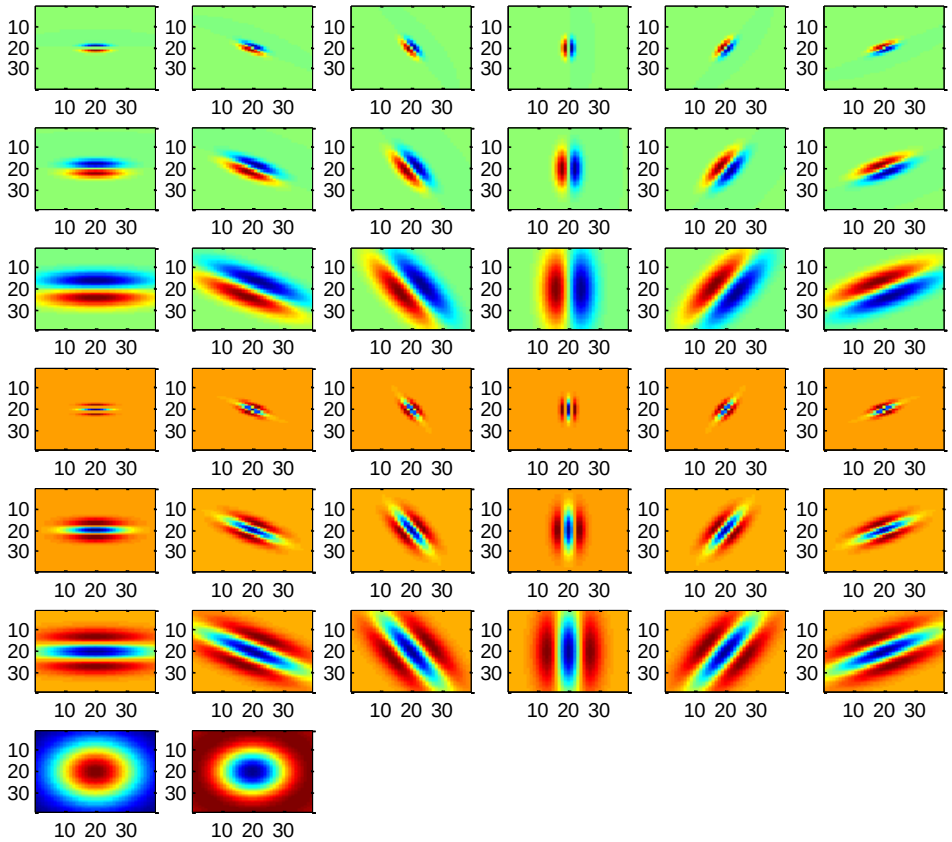

图 3.4 RFS 滤波器. 最后 2 个为高斯与 LOG 滤波器, 前 3 行与中间 3 行分别为边缘和条形滤波器.

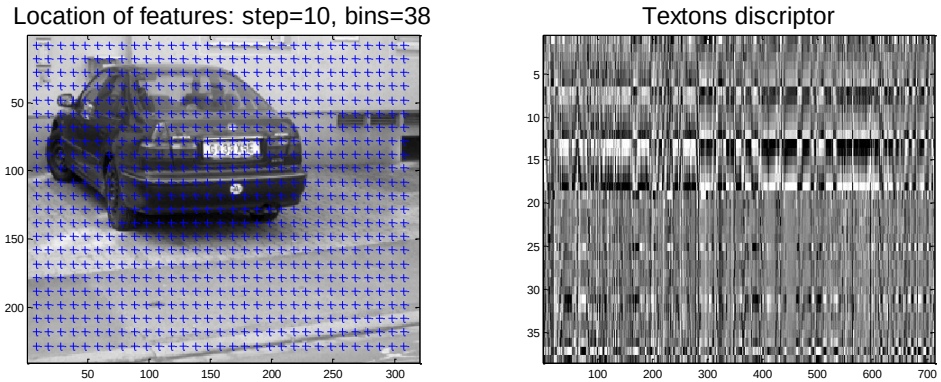

图 3.5 Textons 特征. 左图为 Textons 特征的位置，右图为 Textons 特征描述符.

3.2 超像素获取

超像素的提取是为了能将点特征的描述转化为区域特征[13]描述的基础，超像素就是对图像进行分割且在分割结束后为每个分割区域分配唯一的标签的结果[2]。本文在设计过程中考虑到对数据可扩展性的要求，提供了 QuickShift[14]和 SLIC[15]两种提取超像素的方法，两种方法各有优劣势，总体来说 SLIC 相对 QuickShift 具有更加均匀的分割区域，但颜色一致性较 QuickShift 稍差，因此针对不同的数据集应该权衡选择不同的提取超像素的方式。

3.2.1 QuickShift

QuickShift 是一种很好的获取超像素的方法。算法通过先分割 RGB 图像（或任何双通道以上的图像）成多个区域，然后对分割结果中在图片同一区域的像素使用相同的标签进行标记。这种标定了的区域就是超像素，而且同一图片的不同超像素对应不同的标签，相同区域对应唯一相同的标签。

设原图像像素的颜色可表示为 I(x,y)，其维度（通道数）为 d，彩色图像通常 d=3 而灰度图像 d=1，对图像中的每个像素点 p(x,y)，QuickShift 将(x,y，I(x,y))作为 d+2 维向量空间的样本，这样则在样本像素点的描述中将空间信息与颜色信息综合考虑进去了。QuickShift 算法是一种基于模式查找的聚类算法，通常的模式查找聚类算法要计算 Parzen 密度估计

$$P(x)=\frac{1}{N}\sum_{i=1}^{N}k(x-x_i),x\in R^d \tag{3.4}$$

其中 k(x)为核函数，在 QuickShift 中，k(x)为高斯核。因此，QuickShift 中的 Parzen 密度估计可表示为

$$E(x,y)=P(x,y,I(x,y))=\sum_{x',y'}\frac{1}{(2\pi\delta)^{d+2}}\exp\left(-\frac{1}{2\delta^2}\begin{bmatrix}x-x'\\ y-y'\\ I(x,y)-I(x',y')\end{bmatrix}\right) \tag{3.5}$$

在上式中，将 Parzen 密度估计结果 E(x,y)称为 QuickShift 密度估计的能量，能量越接近的像素点属于同一类的概率越大。

考虑到空间信息(x,y)和颜色信息 I(x,y)在不同图像中的重要性是不同的，因此引入一个比例系数 ratio，用于表示颜色信息与空间信息的比重。图像的每个像素都将计算E(x,y)值，然后构建成一棵树，将当前像素点作为父节点（树根），将比当前像素的E(x,y)值更大的临近像素点都作为子节点。在上述基础上，计算父节点和子节点之间的距离，然后确定与父节点具有最小距离的子节点，然后将(x,y)与最小距离的邻接子节点(x’,y’)连接。最小距离计算方法为

$$dist(x,y)=\min_{E(x',y')>E(x,y)}((x-x')^2+(y-y')^2+ratio*\|I(x,y)-I(x',y')\|_2^2) \tag{3.6}$$

到此为止，将原图像的树结构描述转化为无向带权图（unweight graph），我们给定一个距离的阈值 max_dist，就可以将图分割了，从而形成多个超像素。

同大多数分割问题一样，参数的选择对超像素分割效果影响很大，通过前面的分析，这里总结各参数选取策略：（1）通过观察样本图像，若图像的分割区域颜色一致性比较好，则可以增大 ratio 的取值，使眼色一致性的比重增加；（2）sigma 值太小时，则考虑到的相邻像素的权重太小，这样容易产生过分割，此时因增加 sigma 值，若分割区域过大则应该适当减小 sigma 值；（3）max_dist 参数在极端情况下，设置成无穷大，整幅图片分割成一个超像素，设置成 0 时，每个像素点会分割成一个单独超像素，因此当发生分割区域过大时应该减小 max_dist，出现过分割时应该适当增大 max_dist。实际使用过程中需尽量找到三个参数之间的平衡。

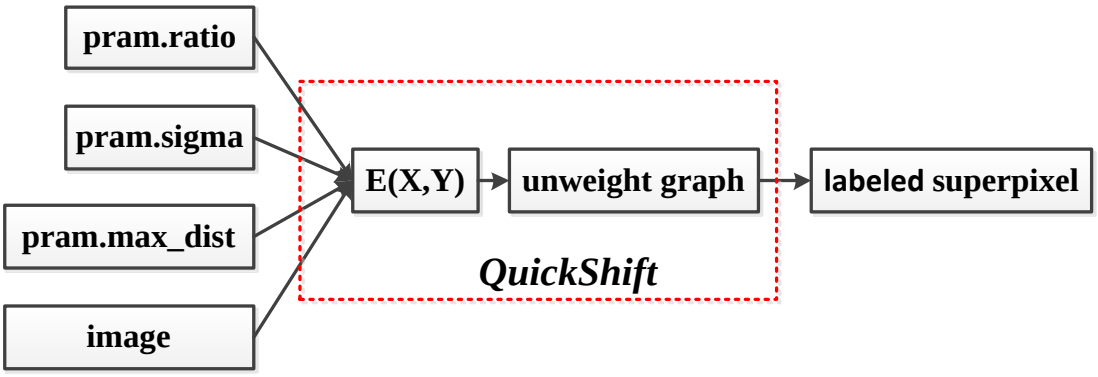

图 3.6 QuickShift 的简单流程框图

QuickShift 产生超像素的优点是算法简单，复杂度较低，分割的颜色一致性较好；缺点是对于颜色复杂的图像，容易产生过分割，常常会出现一个像素分割成一个超像素

的情况，由于 QuickShift 不具备自学习参数的能力，因此人工参数的选择对 QuickShift 的分割结果影响较大。

3.2.2 SLIC

SLIC 是一种简单有效的方法，将一幅图片分割成多个均匀的区域，它的实现基于局部空间的 k-Means 聚类算法。SLIC 像素特征的描述方式与 QuickShift 有类似之处，它对每个像素也使用一个 5 维特征向量描述

$$\Psi(x,y)=\begin{bmatrix}\lambda x\\ \lambda y\\ I(x,y)\end{bmatrix} \tag{3.7}$$

$I(x,y)$ 为像素点在 CIELAB 空间的颜色值，系数 λ 用于平衡位置信息和颜色信息的比重，SLIC 以 K-Means 聚类算法为基础，该特征将用作 K-Means 聚类的特征向量。通过 K-Means 聚类，SLIC 能将具有相似的特征描述向量的像素聚集，但实际过程并不是这样简单，因为 K-Means 算法没有考虑到像素点之间的空间约束关系，所以 SLIC 的 K-Means 算法是被限定在一定物理空间范围内被使用的，称为局部空间的 K-Means 算法（spatially localized k-means）。

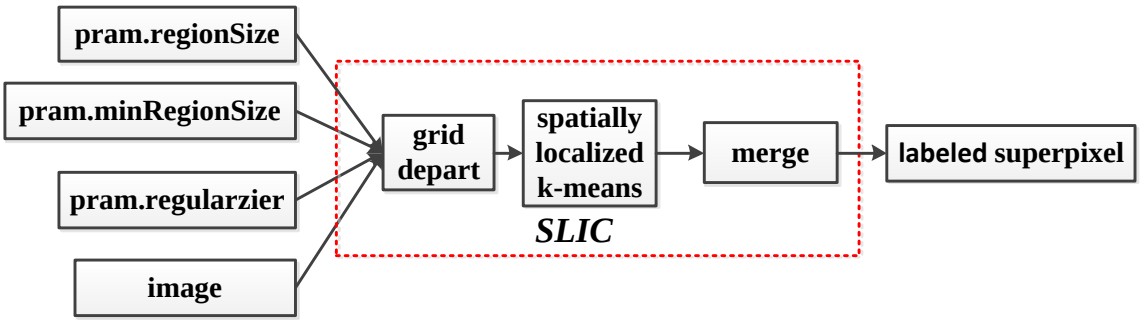

图 3.7 SLIC 的简单流程框图

SLIC 至少需要提供 2 个输入参数：超像素区域的大小（regionSize）和空间规则化的强度（regularizer）。图像首先以 regionSize 为步长分割成一个个均匀网格区域，用每个网格区域的中心初始化 k-means 的聚类起始点。然后 k-means 算法通过 Lloyd 算法迭代产生聚类区域，迭代收敛后同一聚类区域的像素点就形成一个超像素。在计算 $\Psi(x,y)$ 时，设置系数

$$\lambda=\frac{\text{regularizer}}{\text{regionSize}} \tag{3.8}$$

另外，如前所述，在迭代的过程中，考虑像素点之间的空间约束，像素点只能被聚类到2x2 的区域内的 K-means 中心点上。聚类结束后，还需要将区域大小小于minRegionSize 的区域合并到其它区域中。SLIC 算法流程用伪代码可表述为图 3.8。

SLIC 的参数相比于 QuickShift 只有两个，regionSize 直接决定了超像素的数量，regionSize 越大超像素数量越多（或超像素越小），在本文的框架设计中，严格保持这样一种原则：测试过程中，在保证观察者观测到的超像素与实际图像的区域对应较好的情况下，超像素区域越大（或数量越少）越好。regularizer 参数太大将会过分的强化像素点的位置信息，根据公式 3.8 知，其值的选择应该参考 regionSize 的大小而限定在一定范围内。

---

1. Initialize centers $C_k=[x_k,y_k,l_k,a_k,b_k]^T$ by sampling pixels at regular grid steps regionSize.

2. Perdurb cluster centers in an nxn neighborhood, to the lowest gradient position.

3. Repeat unitil converage {

for each cluster center C_k **do**

Assign the best matching pixels form a 2 regionsize x 2 regionsize square neighbohood around the cluster center according to distance measure

$$dist(x,y)=(\lambda\sqrt{(x-x')^2+(y-y')^2}+\left\|I(x,y)-I(x',y')\right\|_2^2)$$

end for

}

4. Enforce merging: superpixels with regionsize<minRegionSize will be merged.

图 3.8 SLIC 算法流程图

3.2.3 对比 QuickShift 和 SLIC

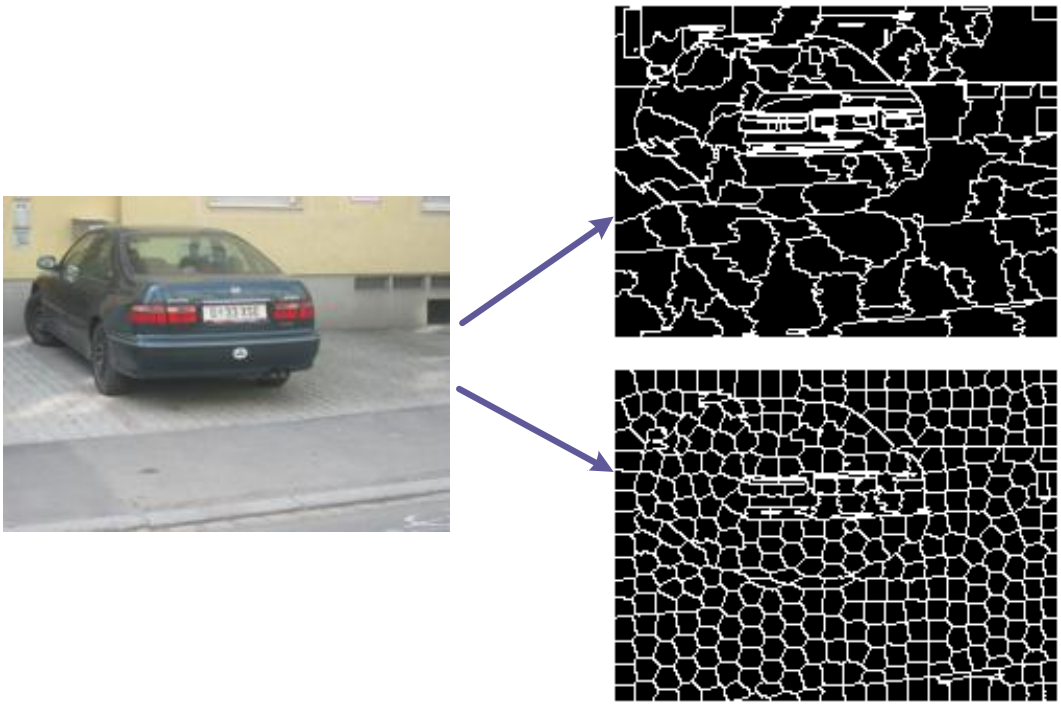

图 3.9 QuickShift 与 SLIC 分割效果对比. 原图像（左），QucikShift（右上），SLIC（右下）.

如图 3.9所示，QuickShift 分割的区域更能保证颜色的一致性，虽然很多区域已经分割较好，但还是存在过分割，这对特征统计是不利的，SLIC 分割的区域大小一致性比较好，但颜色的一致性比 QuickShift 略差。从总体上来说，对于具有复杂背景的图像，SLIC 更适合作为大部分情况下的分割方案，当然，颜色一致性非常强（这种图像往往背景不会太复杂）的图像则强烈建议用 QuickShift。为使得到更佳的预测效果，本设计中将根据输入数据集可选的使用两种分割方法中的一种。

3.3 特征聚类

当获取了像素点特征后，为生成特征包，需要使用聚类算法对像素点特征进行聚类，特征包将属性更接近的特征分配到同一个类型当中。对不同特征，本文可根据情况选择使用普通的 K-Means 算法及其变种层次整形 K-Means 算法（HIKM）。因为像素点特征的数量非常多，HIKM 在以小的精度为代价而换取时间上的效率是值得的，因此将成为本文的首选。

K-Means 算法通过计算每个点到聚类中心点的距离，并迭代更新聚类中心点，直到算法收敛或达到最大的迭代次数，算法收敛时特征点的聚类中心不再发生变化。K-Means 算法中，输入特征可看做高维聚类空间的一个点，算法的初始中心点是通过随机从已有的特征点中选取的，假设聚类数为 K，则初始时从输入特征点中随机选择 K 个特征，随着迭代的进行，聚类中心将可能不在输入特征点集和当中。

图 3.10给出了普通的 K-Means 算法的伪代码。算法中 1{expr}是指示函数，当 expr 为真时结果为 1，否则为 0。在 K-Means 聚类算法中，有两处是算法能否得出好效果的关键：（1）在步骤 1 中，聚类初始点的选取对算法影响很大，若初始聚类中心选取不合适，不仅影响收敛速度，还会使算法达不到全局最优，本文没有对此做优化，使用随机特征作为聚类初始点；（2）步骤 1 中的 $d(x^{(i)},u_j)$ 是距离函数，是数据的相似性的度量，该距离值越大，说明与当前中心点的相似度越低，本文所使用的距离度量即伪代码中所描述的二范式距离。K-Means 聚类算法的时间复杂度为 O(dnkT)，d 表示数据向量的长度，n 表示数据量，k 表示聚类数，T 表示算法达到收敛的迭代次数。因此，K-Means 实际上是一种比较慢的算法，但在聚类领域却已经算最有效的算法了。在实际实现过程中我们将设置聚类的最大迭代次数，虽然这样不好，但能避免迭代收敛时间太长而导致程序效率明显降低的情况。

有些简单的方法能减少 K-Means 计算量，比如 Elkan 算法[16]。Elkan 算法利用了三角不等式及简单的几何原理，考虑下面 2 种情况下，能避免了大量的距离计算：（1）特征点 x 距离现在中心 c1 非常近，而 c1 距离另一个中心 c2 非常远时，则 x 不可能分配到 c2；（2）如果中心 c1 更新到 c2，则 d(x,c2)将被限制在[d(x,c1)-d(x,c2), d(x,c1)+d(x,c2)]范围内。

1. Initialize cluster centroids $\mu_1, \mu_2, ..., \mu_k \in R^n$ randomly.

2. define distance formulation $d(x^{(i)}, \mu_j) = \left\| x^{(i)} - \mu_j \right\|^2$

3. Repeat until convergence: **{**

For every i, assign trainning data $x^{(i)}$ to the closest cluster centroid μ_j, set

$$c^{(i)} := \arg\min_j \left\| x^{(i)} - \mu_j \right\|^2$$

For each j, update centroids, set

$$\mu_j := \frac{\sum_{i=1}^{m} 1\{c^{(i)} = j\} x^{(i)}}{\sum_{i=1}^{m} 1\{c^{(i)} = j\}}$$

}

图 3.10 K-Means 算法伪代码

其实，实际上影响 K-Means 聚类算法时间复杂度的量主要是数据量 n，对于大数据量，更本质的想法是减少 K-Means 中每次迭代时的数据量，层次 K-Means 聚类（HIKM）[18]的方法就是在逐层往下聚类过程中减少聚类的数据范围。层次整数 K-Means 算法建立于 K-Means 算法基础之上，不同的是，为了加快运算速度，在提取特征时对 DenseColor 和 DenseSIFT 以及 Textons 特征描述子进行 0 到 128 的整数归一化处理，对于这些特征，这种处理不会丢失太多的特征信息，之后使用层次的调用 K-Means 聚类算法对其进行聚类，这是一个层次分解的过程。下面我们着重讨论 HIKM 算法的数据结构的描述及其实现过程。

层次 K-Means 聚类的过程可以用一颗倒生长的树描述，HIKM 会随机地选择数据集中的 k 个点作为初始点，每次的层次化都是对数据的一次划分，然后在划分的子集中继续调用 HIMK，直到满足调用的终止条件而退出。HIKM 每一次数据划都是通过在每个节点处调用 K-Means 聚类算法实现的。可以使用 C 代码中的结构体描述 HIKM 树，如图 3.11。

```
struct VlHIKMTree {
    int  M ;                /* 数据向量维度 */
    int  K ;                /* HIKM中迭代调用的K-Means的聚类数 */
    int  max_niters ;       /* 最大迭代次数 */
    int  method ;           /* K-Means计算方法：Lloyd或Elkan */
    int  depth;             /* HIKM树的深度 */
    VlHIKMNode * root ; /* 树根节点 */
};
```

图 3.11 HIKM 树结构的 C 结构体描述

既然 HIMK 可以用树结构描述，因此很容易使用递归实现 HIKM 算法，递归的每个层次都直接调用整数类型的 K-Means 算法。但有一个问题，HIKM 算法的递归终止条件是什么？一种办法是设置树叶数的最大值，即将 nleaves<max_nleaveas 作为递归保持的基准条件，当满足 nleaves>=max_nleaveas 时 HIKM 算法结束。

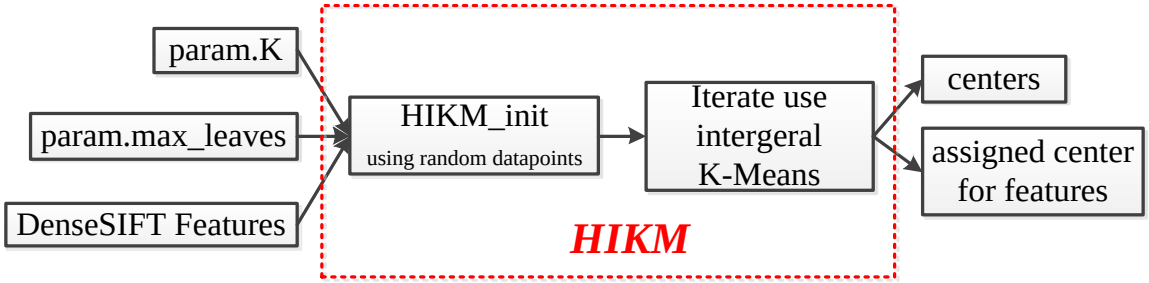

图 3.12 HIKM 简单流程框图

图 3.12是 HIKM 的简单流程框图，聚类的结果中包括聚类中心（centers）以及为每个特征向量被分配到聚类中心的分配编号，编号是通过从树根往树叶每个层次被划分到的聚类中心的顺序描述的，因此编号不是单指一个数值，每个编号都是由层次数个数值组成。HIKM 不是通过 param.K 来描述最终的聚类数目，param.K 表示的是每一层聚类的数目。HIKM 通过从树根到树叶的一条路径表示一类的中心，所以 HIKM 的实际聚类总数是树叶的数目。HIKM 树的深度为$\lfloor \log(K*max_nleaves) \rfloor$。

像素特征聚类的结果就形成特征包[17]（Bag of Features），特征包将用于后续的直方图特征统计中。

3.4 统计直方图与多特征融合

建立超像素统计直方图的过程是：对特征分别进行聚类（设聚类数为 K）获得特征包（Bag of Features），统计每个超像素范围内的特征分布在不同类（1 到 K 类）的特征点数，形成能够描述超像素区域的直方图特征。这样，我们就能通过直方图特征将点的特征转化为超像素的特征，这是从点特征到区域特征的转换。

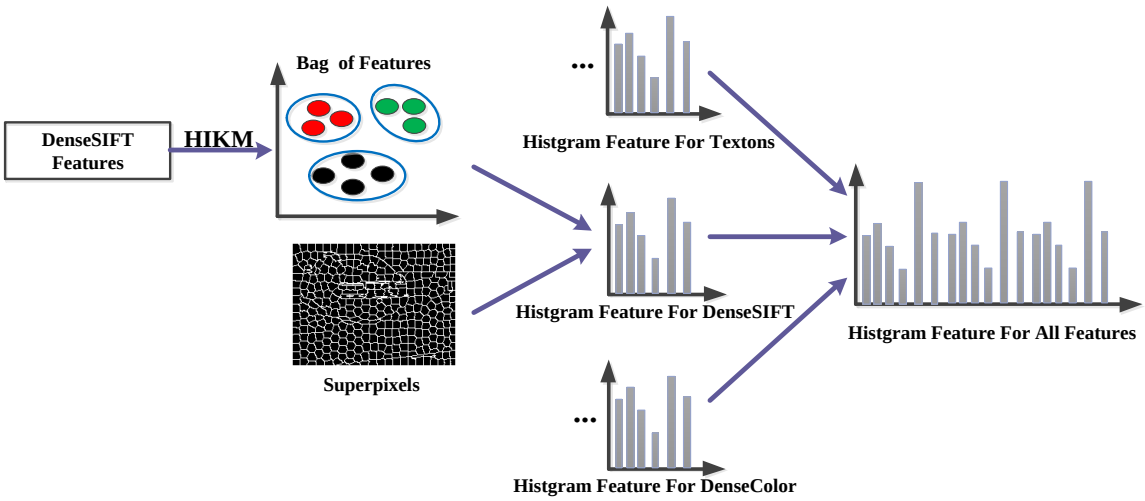

图 3.13 超像素直方图的建立过程

前面已提及了 3 种特征，当然，我们可以只是用其中一种特征，但为了增加预测精度，多种特征融合才更具优势，这点将从本文后面的实验数据分析中看到。当识别图片中的存在旋转与尺度变化的相同目标时，或纹理发生变化时，最好融合 DenseSIFT 与 Textons 特征，本文后面的实验部分将对此做对比的验证。当要使用多种特征时，我们需要将特征直方图组合（如图 3.13），我们的组合方法是

$$H=[\omega_1 H_1,\omega_2 H_2,...,\omega_k H_k] \tag{3.9}$$

其中 Hi 表示第 i 种超像素特征，H 表示组合后特征，[A,B]表示将 A 与 B 向量串联操作，很明显，这将增加组合后超像素特征的维度。并联的方法中使用[A;B]，很明显串联方式但相对于使用并联的方法，数据遗失更少，而且只需要满足超像素个数相同就能

组合，并联的方法则要求满足特征维度必须相同，因此，比较而言，在本文中串联方法更可取。

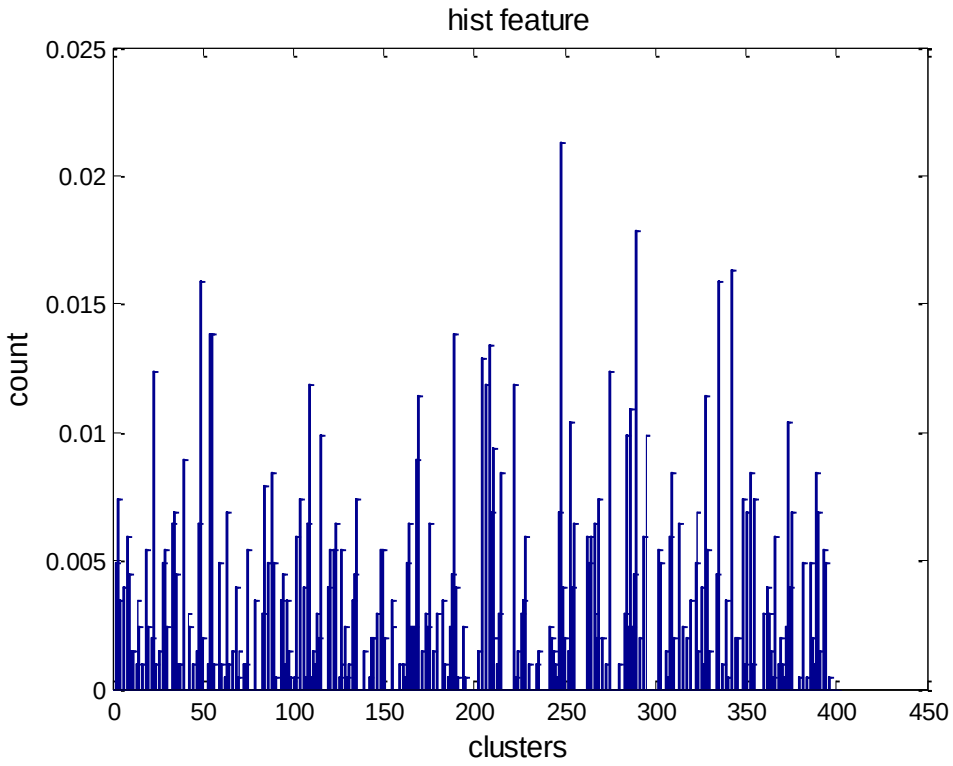

图 3.14 从样本数据中随机抽取的一个超像素直方图特征. 横坐标表示特征维度（K=400）.

前面我们一直考虑的是特征的内部属性，但没有考虑超像素之间的空间关联属性，使用超像素的邻域是考虑空间属性最直观的方法[2]。另一种情况，在聚类数目非常大时，超像素的直方图特征将显得特别稀疏。基于上述原因，我们将超像素的直方图特征更新为

$$H_{ap} = \frac{1}{k+1}(H_{bp} + \sum_{i=1}^{k} w_i H_{bp}(\oplus an)) \tag{3.10}$$

其中等式右边的H_{bp}表示更新前的超像素直方图特征，k 表示H_{bp}的相邻超像素个数，an 表示相邻的超像素的距离，当相邻超像素距离小于等于 an 时都将考虑到求和式当中，直接相邻时 an=1。等式左边的H_{ap}表示添加邻域特征直方图更新后的特征，$\oplus$符号表示满足 an 相邻的关系。等式将相邻的超像素直方图特征考虑在内，更新后的直方图特征

添加了空间信息，但需要注意权值 w_i 的选择，其值过大将导致邻域超像素特征掩盖超像素的内部特征，反而会使效果降低，因此普遍做法是让 w_i 选择一个较小的值。

4 支持向量机

4.1 支持向量机介绍

支持向量机（Support Vector Machine, SVM）是在 1995 年提出的，支持向量机的基本思想是：在样本空间构造最优超平面，使得超平面与不同类样本之间的距离达到最大，错分率最小，从而达到最大的泛化能力和推广能力。错分率最小指样本被错分的概率最小。支持向量机中有两个重要的模型——优化模型和决策模型，优化模型提供了要优化的目标的公式化描述，决策模型的结果返回判决值，判决值用于直接判决样本数据的类属性，即属于哪一类，比如两类的分类问题，若决策模型的结果为正表示第一类，为负则表示第二类。支持向量机只是一种分类方法，本文中用其分类前景目标超像素和背景超像素。超像素通过直方图特征进行描述，因此只需要对超像素的直方图特征分类，从而确定超像素所属的类别，达到目标识别的目的。

在不做特别说明时，本部分中将超像素、超像素特征与样本数据这几个概念混用。用支持向量机解决本文中的超像素分类的问题可描述为：给定训练数据集中超像素的直方图特征 $h_i, i=1,...,l$ ，其中 l 表示超像素的个数，考虑简单的两类线性分类问题，定义指示函数

$$y_i = \begin{cases} 1, & \text{if } h_i \text{ is foregroud} \\ -1, & \text{if } h_i \text{ is background} \end{cases} \tag{4.1}$$

y=1 时的 h 称为正样本，y=-1 时的 h 称为负样本，我们将 y 称为数据标签，简称标签。求解的目标就是对于每一个 h 特征，考虑给定一个 y 值，不同的 y 值决定不同的类，从而使属于不同类的 h 分开，并且这个过程中我们要确保 h 特征被错分的概率尽可能小。支持向量机使用超平面来分开两类数据，超平面是关于样本空间的线性函数，二维为直线，三维为平面，设线性超平面方程为 $w^T h + b = 0$ ，则满足

$$\begin{cases} w^T h + b > 0 \text{ if } y_i = 1 \\ w^T h + b < 0 \text{ if } y_i = -1 \end{cases} \tag{4.2}$$

时超平面能将两类数据分开，但这仅仅能分开，并不能保证错分率最小。进一步将上述式子合并，我们得到两类分类问题决策模型的函数表达式为

$$f(x)=\mathrm{sgn}(w^{T}x+b) \tag{4.3}$$

其中的 w 与 b 都是需要设定的超平面系数，w 表述了其倾斜程度，b 表示超平面相对坐标原点的距离。如图 4.1，通过对超平面执行平移旋转操作，存在一个位置，使两类样本之间的距离最大，这时候达到错分率最小。为进一步求最优的超平面，支持向量机定义了最大余裕概念，最大余裕指超平面正好处在这样一个位置：正负样本能分开且错分概率最小，从几何上描述就是与正负样本相切的超平面之间距离最大，分割超平面在正负样本超平面的正中间。正样本相切超平面为 $w^{T}h+b=1$，负样本相切超平面为 $w^{T}h+b=-1$，两者之间距离为 $d=2/\|w\|$，则最大余裕为

$$\max\frac{2}{\|w\|}\Leftrightarrow\min\frac{1}{2}\|w\|^{2} \tag{4.4}$$

我们将其转化为标准的二次规划问题，最终我们只要求解目标

$$\begin{aligned}&\min_{w,b}\frac{1}{2}w^{T}w\\&\text{s.t. } y_{i}(w^{T}h_{i}+b)\geq 1,i=1,...l\end{aligned} \tag{4.5}$$

就能达到最佳的分类效果，公式 4.5 即为支持向量机的优化模型。

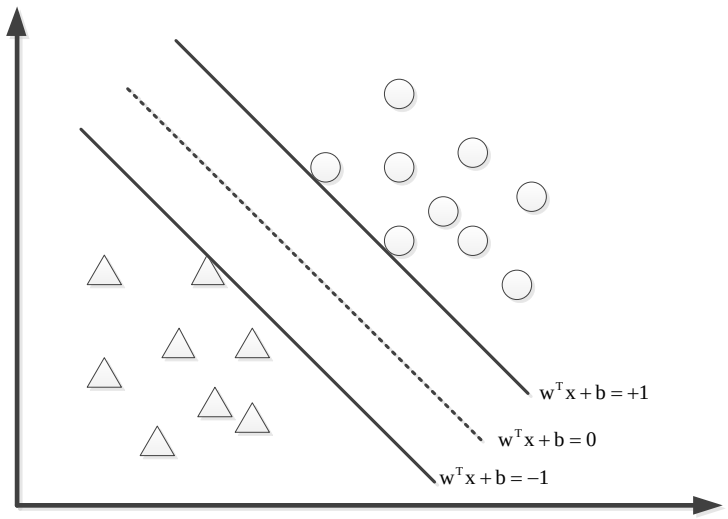

图 4. 1 使用超平面分开两类数据. 虚线为超平面.

原始的优化模型在训练过程中会出现一些问题，比如过拟合（over-fitting），为防止过拟合，增加支持向量机的鲁棒性，因此在优化模型中添加惩罚项$C\sum_{i=1}^{l}\xi_i$，惩罚项能够在过拟合发生时抵销过拟合的成分，是一种补偿的效果。因此，前面描述的标准的线性支持向量机的优化模型变为

$$\begin{aligned}&\min_{w,b}\frac{1}{2}w^Tw+C\sum_{i=1}^{l}\xi_i\\&\text{s.t. } y_i(w^Th_i+b)\geq 1-\xi_i\\&\xi_i\geq 0,i=1,...l\end{aligned}\tag{4.6}$$

其中 C 为惩罚项系数。

以上讨论的都是基于训练数据可以使用线性超平面分开的支持向量机，而实际问题中许多数据是线性不可分的，为此，引入核函数。核函数只是做了一个映射，将低维空间线性不可分的数据映射到高维空间，在高维空间中，特征的维度增加，倘若映射函数选择恰当，则能够将不同类型的数据映射到最高维的不同尺度区域上，然后在高维空间的超平面中实现线性可分，使用线性支持向量机模型。我们将核函数放到标准线性支持向量机的约束条件当中，因此得到一个比较通用的支持向量机模型，优化模型为

$$\begin{aligned}&\min_{w,b}\frac{1}{2}w^Tw+C\sum_{i=1}^{l}\xi_i\\&\text{s.t. } y_i(w^T\Phi(h_i)+b)\geq 1-\xi_i\\&\xi_i\geq 0,i=1,...l\end{aligned}\tag{4.7}$$

决策模型为

$$f(x)=\text{sgn}(w^T\Phi(h)+b)\tag{4.8}$$

优化模型与决策模型中的$\Phi(h_i)$都是指新添加的核函数。一个比较好用的核函数为 RBF 核函数，其表达式为

$$K(h_i,h_j)=\exp(-\text{gamma}*\|h_i-h_j\|^2)\tag{4.9}$$

gamma 为映射系数，h_i与h_j表示的是超像素特征的直方图描述。在数据量比较小或者适中时应该尝试使用 RBF 核函数，但对于大数据，使用线性核函数将获得时间上效率

的很大提高。本文在小量样本时使用 RBF 核函数支持向量机，在大数据样本（比如 Graz-02 数据集）中将更多的使用下面的线性分类器的支持向量机，

$$\min_{w}\frac{1}{2}w^{T}w+C\sum_{i=1}^{l}(\max(0,1-y_{i}w^{T}h_{i}))^{2} \tag{4.10}$$

称之为 L2-regularized L2-loss SVC 分类器，线性分类器在大数据时因为不用使用到高维空间的映射操作，因此能节省大量的时间和内存空间。

之前讨论的都是两类分类问题，对于多类分类的问题，可使用了两种机制：一种是一对其余，另一种是一对一。一对其余是指选择其中一类为正样本，其余都被归为负样本，先将归为正样本的类分开，依此类推，再选择另一类为正样本再分开。一对一指从多类中选择两类分别作为正样本与负样本，先将这两类分开，依次迭代将每两类都做依此分类，假设有 C 类，则共需要做 C(C-1)/2 次两类分类。很明智的，支持向量机通过上述两种方式将多类分类问题转化为两类分类问题。

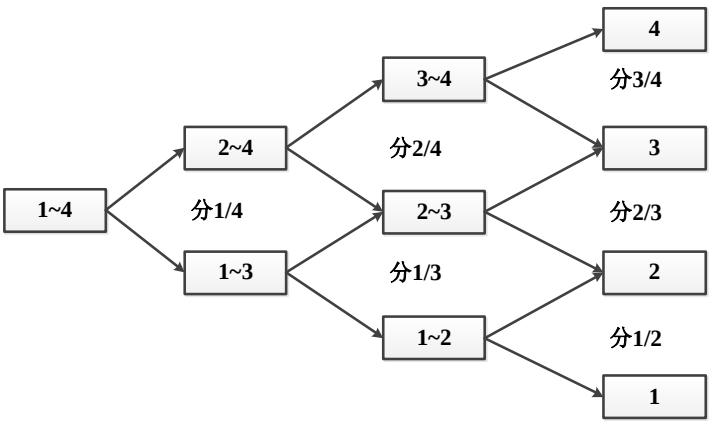

图 4.2 支持向量机中的多类分类问题. 一对一的分类方式.

4.2 支持向量机的参数优化

支持向量机的参数对分类效果的影响非常大。因此，参数优化是必不可少的一步，支持向量机中的参数优化问题就是如何选择最优的参数 gamma 和 C 使模型的泛化特性最佳。优化模型中的惩罚项是为了抵消过拟合而添加的，惩罚 C 太小，容易产生过拟合，C 太大，优化模型过于平坦，噪声大于有效数据，反而更加糟糕。因此 C 的选择是：保证不发生深度过拟合的情况下，尽量选择比较小的 C 值。若使用 RBF 核函数的

支持向量机，则核函数中的 gamma 直接影响到低维空间到高维空间的映射效果，gamma 值取得好，能很方便的在高维空间上线性可分。

参数优化之前必需确定优化性能的衡量指标。本文使用交叉验证（Cross Validation, CV）思想来评估分类器性能。交叉验证技术是用来验证分类器性能的一种统计方法，基本思想是将样本数据分成多个组，首先从中选择任意一组组作为验证集，其它各组（剩余数据）都用作训练集，使用训练集对分类器进行训练，再利用验证集测试训练得到的模型的精度，然后再依此选择其它组为验证集，剩余数据为训练集测试模型精度，最后计算平均精度，以此来作为评价分类器的性能指标。假设超像素特征共 N 个，交叉验证均分为 K 组，则每组数据有 N/K 个超像素特征，将每个子集分别做一次验证集，其余 K-1 组子集数据作为训练集，这样会得到 K 个模型，则使用用这 K 个模型最终的验证集的分类准确率（Accuracy）的平均数作为评价支持向量机分类器的性能指标，分类准确率的定义为

$$\text{Accuracy} = \frac{\text{true\_all}}{\text{all}} \tag{4.11}$$

true_all 表示预测正确的样本数，all 表示样本的总数。

参数优化方法上，本文在设计过程中曾研究过多种优化方法，其中以格点搜索（girdSearch）和遗传算法优化为主。考虑到遗传算法的局部优化特性，且遗传算法本身参数选取不当很难找到最优的效果，因此实际实现过程中使用格点搜索的方法。格点搜索方法设定参数的范围与步进值，所有的可能的参数取值序列构成格点集合，比如 C 与 gamma 参数构成集合 $\{[C_1,\text{gamma}_1],[C_1,\text{gamma}_2]...[C_k,\text{gamma}_k]\}$ 。参数集中的每一对参数都作为支持向量机的输入，然后使用交叉验证评估每组参数的平均分类准确率，以分类准确率最高的一组参数作为最终的参数优化结果。

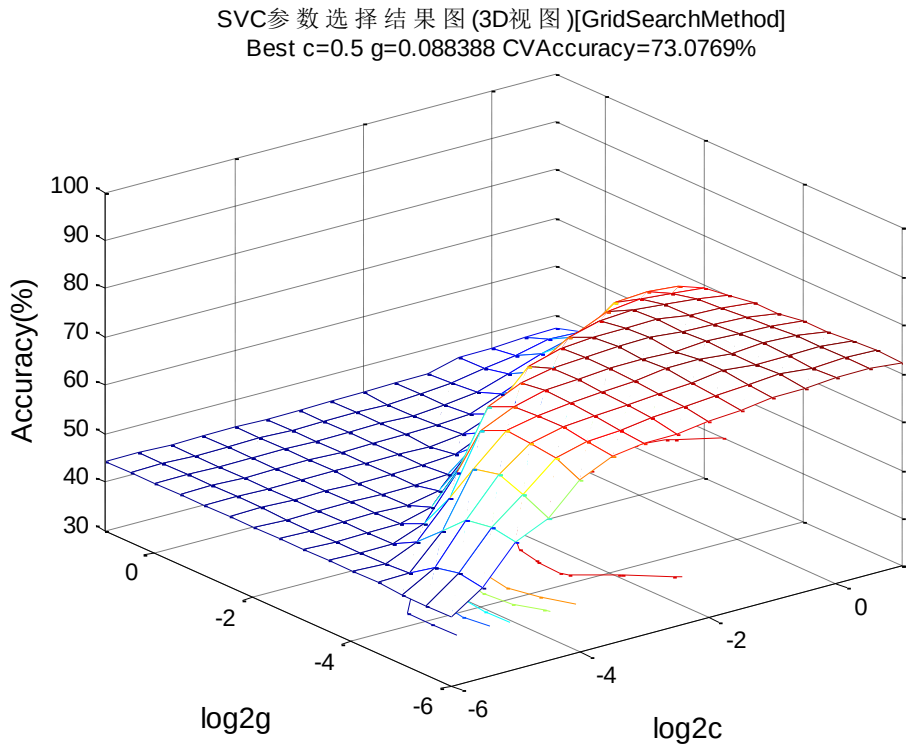

图 4.3 使用 RBF 核函数的交叉验证. 从 graz-02 数据集中选取了部分样本，对 gamma 与 C 参数的使用格点搜索方法优化.

对于采用线性分类器分类的问题，最重要的需要优化的参数使 C，而对于使用 RBF 核函数的支持向量机，则至少需要尝试对 gamma 与 C 优化。

4.3 使用支持向量机的非平衡样本数据问题

garz-02 数据集中存在样本数据不平衡的情况，以其中的 car 数据集为例，前景获得的超像素与背景超像素数量的比例是 1:9。按照支持向量机使用超平面分类的原理，不平衡数据不会对分类效果造成影响，但不平衡样本数据会对惩罚项产生影响。因此，需要修改惩罚项的格式，将训练样本的类型考虑在其中，使用 $C_+\sum_{y_i=1}\xi_i + C_-\sum_{y_i=-1}\xi_i$ 替换之前的惩罚项，C+与 C-分别表示用于正样本惩罚项与负样本惩罚项的系数。

除了对惩罚项修改之外，对于不平衡样本数据，使用分类准确率作为交叉验证中的评估标准不再是很好的想法。比如，考虑这样一种极端化情况，负样本远大于正样本，当所有的样本数据的预测结果为负样本时，实际这种预测结果是很糟糕的，但却能得到

很高的分类准确率。因此，本文改使用分类平衡精度（Banlanced Accuracy, BAC）作为分类效果的评估标准，平衡精度的定义为

$$\mathrm{BAC}=\frac{\mathrm{Sensitivity}+\mathrm{Specificity}}{2} \tag{4.12}$$

$$\mathrm{Sensitivity}=\frac{\mathrm{true\_positive}}{\mathrm{true\_positive}+\mathrm{false\_negative}} \tag{4.13}$$

$$\mathrm{Specificity}=\frac{\mathrm{true\_negative}}{\mathrm{true\_negative}+\mathrm{false\_positive}} \tag{4.14}$$

其中 BAC 表示平衡精度，true_postive 表示预测结果中正样本预测正确的标签数，false_negative 表示预测结果中负样本预测错误的标签数，true_negative 表示预测结果中负样本预测正确的标签数，false_positive 表示预测结果中正样本预测错误的标签数。平衡精度 BAC 具有平衡对称特性，不依赖于正负样本数的差异，因此可代替分类准确率 Accuracy 用于非平衡数据在交叉验证中的评估标准。

4.4 支持向量机的使用步骤

支持向量机的在使用过程中除了要仔细考虑前面提到的参数优化，非平衡数据等问题之外，一个仔细地对样本数据的预处理对预测结果影响也非常大。因此，在一定程度上遵循支持向量机的使用步骤能帮助更好的使用支持向量机获得更好的效果。

在本文中优先考虑的支持向量机的使用步骤为：（i）从原始图像中提取超像素特征，从掩码图片中获取超像素标签信息，运用统计的方法，将落在超像素区域类像素点最多的一类作为超像素的标签类；（ii）考虑使用简单的数据归一化操作对样本数据预处理，对于如本文的非负数据，将数据归一化到[0 1]是一种很好的选择；（iii）对于小样本或中等样本数据，优先考虑使用 RBF 核函数类型的支持向量机，虽然 RBF 核函数训练时间比线性的支持向量机长，但其将数据映射后的超平面划分效果将会很好；（iv）使用交叉验证获取最佳的参数，其中要使用一个好的评估标准，在使用非线性核函数时，交叉验证需要获取 gamma 和 C 两个参数组合的最佳值，在线性支持向量机里最重要的参数值是 C，因此应该使用交叉验证获取最佳的 C 值；（v）使用测试数据集

对训练模型测试，前面提到，我们将样本数据做了划分，其中一部分将用于模型的训练，余下的部分用于验证模型的效果，验证模型的过程就是用测试数据集测试的过程。

5 条件随机场

通过支持向量机获得的模型已经能得到一个较好的识别效果了，但支持向量机识别结果更多的考虑的是超像素本身的信息，而且支持向量机模型产生的目标边界不平滑，使用条件随机场（CRF）可以使解决这些问题。

5.1 条件随机场介绍

我们给条件随机场下一个定义：设 G=(V,E)是一个无向图，$Y=\{Y_v \mid v\in V\}$ 是以图 G 中节点 V 为索引的随机变量Y_v 构成的集合，Y_v 叫输出变量，因此 Y 也成为输出变量集合。在给定输入变量集合 X 的条件下，如果每个输出随机变量Y_v 服从马尔可夫性，即 $p(Y_v \mid X, Y_u, u\neq v)=p(Y_v \mid X, Y_u, u\oplus v)$，其中 $u\oplus v$ 表示邻接关系，则(X,Y)构成一个条件随机场。其中图 G 定义为$p(y \mid x)=\frac{1}{Z}\prod_A \Psi_A(y_A, x_A)$，$\Psi_A(y_A, x_A)$ 称为势函数。

图 G 定义中的概率描述的观察序列发生的概率，一个最基本的原则是使以基准事实为观测序列的概率达到最大，即 max{p(y|x)}，这是条件随机场的求解目标。本文使用 s 表示超像素，c 表示分配给超像素的标签，给定条件随机场中的势函数为

$$\Psi(c_i, s_i)=\exp\{-\sum_{s_i\in S} w_1 U(c_i \mid s_i)-\sum_{(s_i,s_j)\in E} w_2 V(c_i, c_j \mid s_i\oplus s_j)\} \tag{5.1}$$

为求 p(y|x)最大，可转化为求能量函数 En 最小值[22][23]，即

$$\min E_n=\min\{\sum_{s_i\in S} w_1 U(c_i \mid s_i)+\sum_{(s_i,s_j)\in E} w_2 V(c_i, c_j \mid s_i\oplus s_j)\} \tag{5.2}$$

其中 U 描述的是条件随机场中个体内部特征，称作一元势项（unary-term），V 描述了相邻个体之间的关联关系，称作二元势项（pairwise-term），w1 与 w2 是权值系数。为此，要使用条件随机场，有 3 个待解决的问题：（1）定义恰当的一元势项和二元势项，（2）找到一种方法求解 En 的最小值，（3）选择合适的 w1 和 w2 的参数值。

5.2 构造基于超像素的条件随机场

构造条件随机场的首要任务是构造图结构，本文将根据超像素构造图结构，邻接关系使用四邻域描述。节点和边的构造方法：将每个超像素当作图的一个节点，将四邻域相邻的超像素连接形成边，如图 5.1。

除了图结构的构造，我们需要构造一元势项与二元势项，以对图进行精确的描述。我们先设一幅图中的超像素个数为 N，数据标签类数为 C。一元势项代表图结构中节点类型被划分到对应标签的概率，因为图是通过权值描述，所以以分配到对应标签概率越大，一元势项的值越小。这里使用使用支持向量机决策模型的判决输出值来定义一元势项，

$$U(c_i \mid s_i) = -\log(p(c_i \mid s_i)) \tag{5.3}$$

其中的 p 表示支持向量机的决策函数输出。一元项尺寸为 CxN。二元势项定义了图结构中相邻超像素节点的关联程度，直观反映到图结构中就是连接超像素的边的权值。

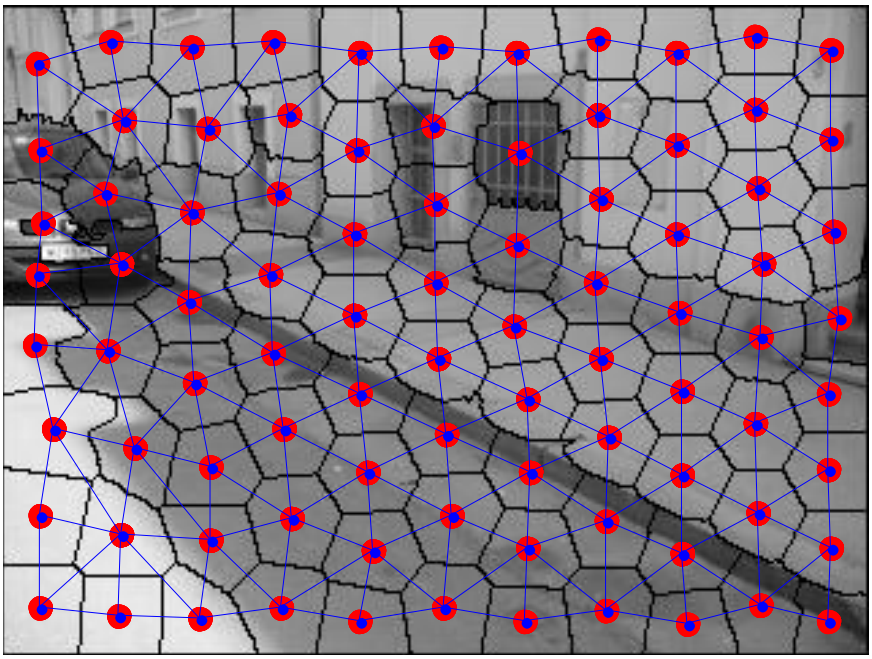

图 5.1 构造超像素数据结构图. 以超像素为节点，相邻的超像素连接起来作为结构图的边.

考虑颜色信息与超像素边界关联信息，本文中将二元势项定义为

$$V(c_i, c_j \mid s_i \oplus s_j) = \left(\frac{L(s_i, s_j)}{1 + \| s_i - s_j \|} \right) * [c_i \neq c_j] \tag{5.4}$$

其中的 L 为相邻超像素之间的公共边界长度（share boundary length），$\| s_i - s_j \|$ 代表了不同超像素之间的颜色差异，

$$\| s_i - s_j \| = \sum_{rgb} [color(i) - color(j)]^2 \tag{5.5}$$

$[c_i \neq c_j]$ 为指示函数，分母中加 1 是为了避免分母为 0 的情况，这并不影响条件随机场的性能。二元项尺寸为 NxN。为方便对条件随机场问题的求解，构造二元势项时必需满足下面的几个条件，

$$V(c_i, c_j) = 0 \Leftrightarrow c_i == c_j \tag{5.6}$$

$$V(c_i, c_j) = V(c_j, c_i) \geq 0 \tag{5.7}$$

条件一要求二元势项矩阵的对角线元素为 0，条件二要求二元势项矩阵是对称矩阵，且元素值都必须非负。本文前面构造的二元项 V 即满足上述条件。

关于能量函数最小值的计算，可以通过图模型转化为计算网络流图的最大流最小割问题。在条件随机场求解过程中，根据预测的数据标签更新的方式不同有 $\alpha - expansion$ 或 $\alpha - \beta$ swap 两种方法，但都使用最大流最小割算法计算能量最小值。

---

1. Start with an labeling {c} which come from svm output

2. Set success := 0

3. assume L is possible label set. For each label $c_i \in L$

(1) Find $\{c^{\wedge}\} = \arg\min En(\{c^{'}\})$ among $\{c^{'}\}$ within one c_i expansion of $\{c\}$

(2) If $En(\{c^{\wedge}\}) < En(\{c^{'}\})$, set $\{c\} := \{c^{\wedge}\}$ and success := 1

4. If success = 1 goto Step 2

5. Return {c}

图 5.2 α-expansion 的算法伪代码

1. Start with an labeling {c} which come from svm output

2. Set success := 0

3. assume L is possible label set. For each label $(c_i, c_j) \in L$

(1) Find $\{c^{\wedge}\} = \arg\min En(\{c'\})$ among $\{c'\}$ within one c_i and c_j swap of $\{c\}$

(2) If $En(\{c^{\wedge}\}) < En(\{c'\})$, set $\{c\} := \{c^{\wedge}\}$ and success := 1

4. If success = 1 goto Step 2

5. Return {c}

图 5.3 $\alpha-\beta$ swap 的算法伪代码

$\alpha-$expansion 算法起始时，将支持向量机的已有的分类判别结果标签作为条件随机场标签序列的初始值，然后$\alpha-$expansion 过程就是将标签序列中的一个标签使用可能的其它标签替换，将替换后的标签序列作为观测空间序列，重新计算能量值最小从而使观测空间发生概率最大，找到对应的标签序列$\{c^{\wedge}\}$，将该标签序列能量值与$\{c'\}$的能量值比较，若新标签序列的能量值更小则更新标签序列，按此迭代直到标签序列不再发生变化，程序收敛。$\alpha-$expansion 时间复杂度为 O(L) 。$\alpha-\beta$ swap 算法的流程与$\alpha-$expansion 类似，唯一不同的是其使用标签交换的方式更新观测空间的标签序列，正因为此，其时间复杂度是$\alpha-$expansion 为$O(L^2)$ 。

$\alpha-$expansion 与$\alpha-\beta$ swap 算法的 3.1 步中计算 minE({c})都是通过最大流最小割算法计算取得，最大流最小割的网络流图基于前面已构造超像素结构图，但除了使用超像素作为网络流图的节点之外，将可能的标签集（两类分类问题标签集只有 2 个元素）也考虑到网络流图的节点范围之内，称之为端节点，而超像素节点称为普通节点，普通节点之间的链路称为邻接链路（n-link），普通节点与端节点之间的链路称为端链路（t-link），端节点之间不存在连接关系，如图 5.4。

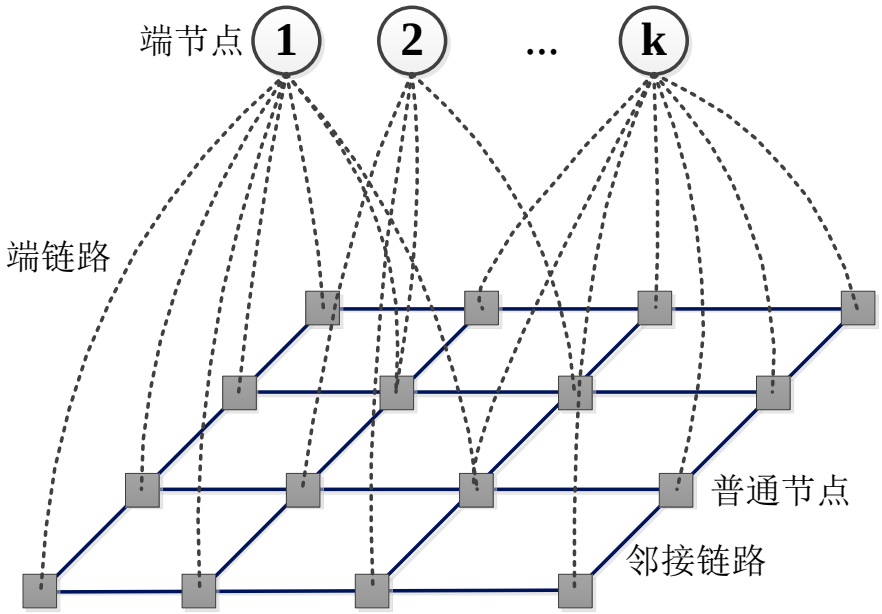

图 5.4 用于计算能量函数最小值的网络流图

在网络流图中，邻接链路的流量使用二元势项表示，端链路的流量使用一元势项表示，不考虑流方向。以两类分类问题为例，标签集为{1,2}，我们要计算能量函数值最小，将 1 设为网络流的起点，2 设为网络流的终点，则目标是寻找一个起点到终点之间的割集，使割集的流量 f 最小，割集是部分端链路与部分邻接链路的构成的集合。

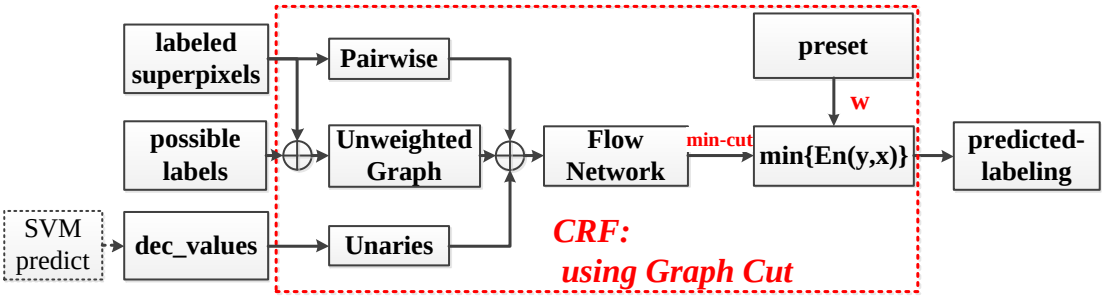

图 5.5 条件随机场的流程框图

条件随机场总结：如图 5.4，条件随机场使用标定的超像素构造二元势项，支持向量机的决策输出值作为一元势项，从而构造流网络。然后最关键的是使用图割算法（最小割）求解能量函数最小值。能量函数中的一元势项与二元势项的权重系数值进行人工预设定。条件随机场的输出是通过是能量函数最小调整后的数据标签，根据数据标签和

标定的超像素很容易以图像的方式建立最终需要的掩码图片。虽然条件随机场中参数的选择很重要，但本文尚未对此做分析，[19]提出了一种基于结构化的参数训练方法。

6 实验

本文框架的实现环境为 matlab 与 C/C++语言混合编程，系统环境为 x86，CPU 为 2.1GHz，内存 2GB。本文将使用到的数据集提取特征后数据量非常大，由于内存环境的限制，我们会采取一些措施应对大数据量的问题，但“时间与空间不可两全”似乎成为一条哲学，因此我们使用的磁盘数据交换，使用循环方法代替直接的矩阵操作等方法都是以时间为代价而降低瞬时的内存使用。

本文中，聚类算法、DenseSIFT 特征点提取等都是 C/C++语言的实现，通过 matlab 编译调用这些模块。本文使用 libsvm[20]与 linearsvm[21]工具箱实现支持向量机，这两工具箱的效果是公认的较好的。使用论文[22][23]中提供的 gco-v3.0 工具箱实现条件随机场，另外将使用 vlfeat 工具箱[25]做一些基本的图像处理，本文的编码过程借鉴了大量的开源社区的经验以及软件工程中的一些优秀开发方法。

对于单目标识别（两类）问题，考虑到需要掩码图像作为基础事实[26]，本文使用数据集 Graz-02 进行测试，Graz-02 数据集包含 3 个子集，分别是 cars、person 与 bikes。本文将分别对这些数据子集进行测试，主要考虑不同的数据集，在使用不同特征组合的情况下，得到不同精度的识别效果的对比。除此之外，本文也将尝试对多目标识别的数据集测试，使用 Sowerby 数据库，Sowerby 数据库包括天空、树、房子、道路等多类景物。

6.1 Graz-02 两类目标识别测试

因为 Graz-02 数据集中图片尺寸较大（640x480），可用的计算机环境在内存及 CPU 性能上都没法满足对原始的所有数据集直接训练，因此实验中将 Graz-02 数据集所有图片尺寸都压缩为原来的 1/5（尺寸变为 128x96），这样，这样从读取图片到特征提取以及训练整个过程的耗时约为 1 个小时，但是，因为将图片缩小是降采样过程，这会在一定程度上降低识别的精度。Graz-02 数据集比较复杂，颜色多样，因此本文尝试使用各种方式的组合特征，超像素的提取方法使用 SLIC，SLIC 分割结果叫均匀，便于做特征统计。点特征提取过程中，将采样间隔（patchsize）都设为 1 个像素，DenseColor 中颜色信息在 RGB 空间获得， Textons 中的 RFS 滤波器最大尺寸取 5。默认每两种特

征之间的权重比为 1:1。超像素提取中设置 regionsize=8，regularizer=0.1，则每幅图片产生的超像素个数平均约为 768 个。从 300 幅图片中随机选择一半用于训练，另一半用于测试，则用于支持向量机训练的特征有 150x768=115200 个。对点特征的聚类，都使用 HIKM 聚类，设置层次分支聚类数为 20，最少树叶数为 50，则实际聚类数为 2^20=400 类。因此，用于支持向量机训练的特征维度最大为 400x3=1200。实验中考虑 3.4 节给出的相邻超像素直方图更新方案，默认设置邻接超像素距离为小于 3（an=3）的直方图考虑到更新范围之内。

支持向量机的选择：使用 linearsvm 工具箱中提供的 L2-regularized L2-loss SVC 分类器，通过交叉验证获得最佳惩罚系数 C。由于条件随机场的参数尚未使用特定的方法进行训练，默认条件下在训练时设置 w1 与 w2 在范围 1~10 的范围内迭代交叉验证，调节到最佳。

表 6.1 graz-02 数据集在只使用支持向量机与使用条件随机场后的测试平衡精度. graz-02 数据集包括 cars/person/bikes 共 3 个子集. 训练使用主要特征为 DenseSIFT 特征，通过特征组合的方式使用了 DenseColor 与 Textons 特征.

Accuracy(%)	SVM			CRF		
	cars	person	bikes	cars	person	bikes
DenseSIFT	67.5708	59.4476	64.6251	**78.9412**	64.6466	67.3962
Textons	61.0917	57.6827	61.9701	63.7758	61.4204	65.5449
DenseColor	62.5828	57.2178	62.5868	67.2242	60.9080	62.5868
Textons+DenseColor	68.0766	57.3037	66.5458	75.0764	66.9004	67.9440
DenseSIFT+DenseColor	63.4438	56.7812	68.8487	69.0546	64.3563	70.1978
DenseSIFT+Textons	62.7583	60.3473	64.8447	68.7106	65.3749	67.7714
DenseSIFT+DenseColor+Textons	64.1373	61.4438	68.4991	68.9824	**68.0969**	**70.4657**

从表 6.1 中可以看出，cars 数据集在只使用 DenseSIFT 特征时效果最好，而 person 数据集和 bikes 数据集在融合 3 种特征时才达到最佳识别率。除此之外，使用条件随机场优化之后的效果，普遍要比支持向量机的训练效果好。条件随机场使分类的一致性得到了提高，目标的边界更平滑，空间一致性更好（如图 6.1）。本文将识别结果与 Fulkerson 等人的结果做了一个对比（表 6.2），从中可以看出，在 cars（+6.7412）和

person（+1.7969）数据集上精度都有所提高。由此可知，合理地使用多特征融合能在一定程度上改善目标识别的效果，但滥用也可能导致性能的下降，这是因为当目标与背景之间存在特征相似性（比如颜色）时，若采用该特征作为训练的一部分，反而会使背景与目标之间发生混淆。另外，条件随机场的结果在于对支持向量机的结果优化（主要是边缘效果），其无法做到本质上的对识别效果欠佳的图像的改变。

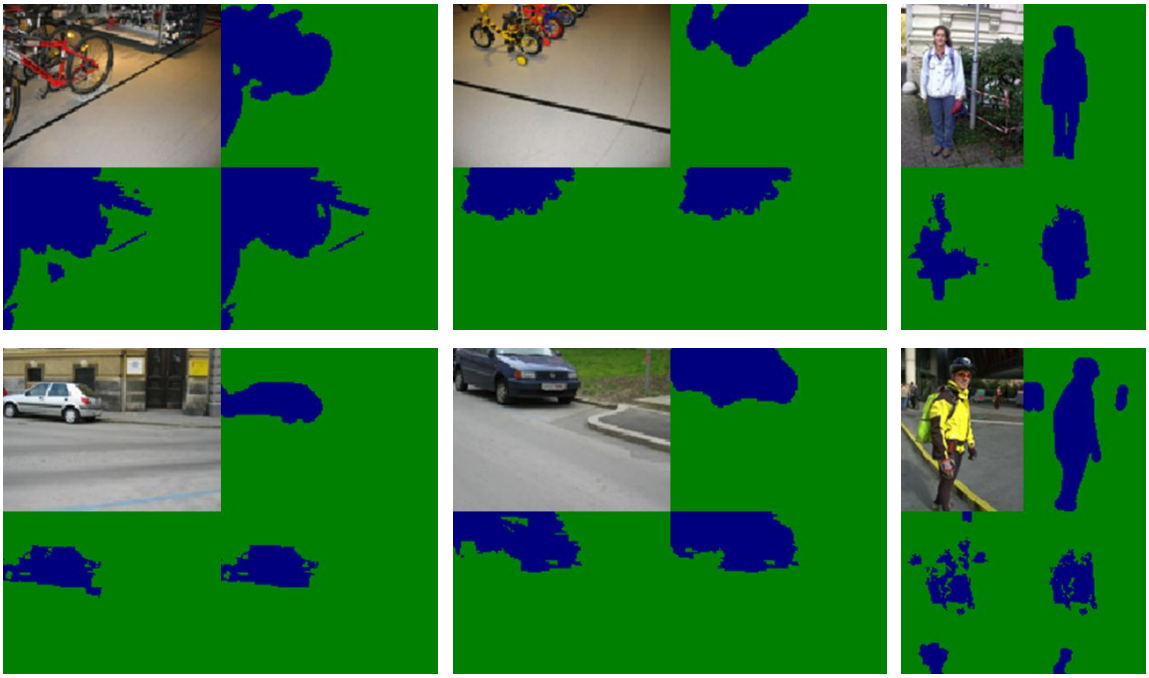

图 6.1 graz-02 数据集预测效果图. 共 4x4=16 幅图片，每个子图代表一个预测结果，每个预测结果从左往右、从上到下依次为原图像、掩码图像、支持向量机预测结果和条件随机场调节后结果.

表 6.2 本文结果与 fulkerson 结果对比（graz-02）

Accuracy(%)	cars	person	bikes
fulkerson[2]	72.2000	66.3000	72.2000
this paper	78.9412	68.0969	70.4657
improved	+6.7412	+1.7969	-1.7343

6.2 Sowerby 多类目标识别测试

Sowerby 数据库中的图片都 比较小，尺寸为 96x64，图片数量共 100 幅。本实验对 Sowerby 数据集分别采用 QuickShift 与 SLIC 提取超像素的方法进行比较，比较使用 DenseSIFT 和 DenseColor 或者两者组合特征时的效果。像素特征采样间隔为 1 个像素，DenseColor 在 RGB 彩色空间提取特征。像素特征聚类也是用 HIKM，设置 K=20，max_nleaves=50，则实际的聚类数为 20^2=400。QuickShift 参数设置为：ratio=0.5，kernelsize=0.5，maxdist=8。SLIC 参数设置为 regionsize=8，regularizer=0.1。QuickShift 对不同图片分割的超像素数量及大小不固定（多则 119 个，少则 60 个，平均约 80 个），而 SLIC 的超像素维持在一个小的波动范围，而超像素个数都一样，采用以上参数 SLIC 的超像素个数约为 96 个。因此将 100 幅图片随机选择 50 幅用于训练，剩余 50 幅用于测试，则针对 SLIC 超像素提取方式，总的用于训练的特征数约为 50x96=4800 个，针对 QuickShift 超像素特征提取方式，总的用于训练的特征有 50x80=4000 个。与 graz-02 一样，对 sowerby 我们也考虑相邻超像素，设置 an=3。即使 sowerby 数据量不大，但考虑到实验硬件条件的约束，本文仍使用 linearsvm 工具箱的线性支持向量机用于支持向量机的训练。在本实验中，对于多类分类的问题，由于支持向量机的应对多类分类采用可能两种不同的机制，条件随机场中一元势项的处理比单目标识别问题中要稍微复杂，本框架尚未对其做相应处理，因此本实验中没有使用条件随机场的优化过程。默认各种特征的权重比为 1:1。

表 6.3 sowerby 数据集的测试平衡精度.

Accuracy(%)	SLIC	QuickShift
DenseSIFT	61.4880	55.0021
DenseColor	64.4018	65.9894
DenseSIFT+DenseColor	**71.4828**	65.8198

表 6.3 可以对比了使用不同超像素提取方式，不同的特征或特征组合的情况下，sowerby 数据集的测试平衡精度。在使用 SLIC 超像素提取方法，DenseSIFT+DenseColor 特征组合的情况下，sowerby 测试精度达到最高（71.4828%），单独使用 DenseColor 的

效果比单独使用 DenseSIFT 的效果要好，这得益于 sowerby 数据集的不同类之间的颜色划分比较清晰。本数据集中，使用 QuickShift 容易发生过分割。一般情况下，使用 SLIC 提取超像素，尝试使用各种不同的特征组合方式，能找到一个较好的识别效果。

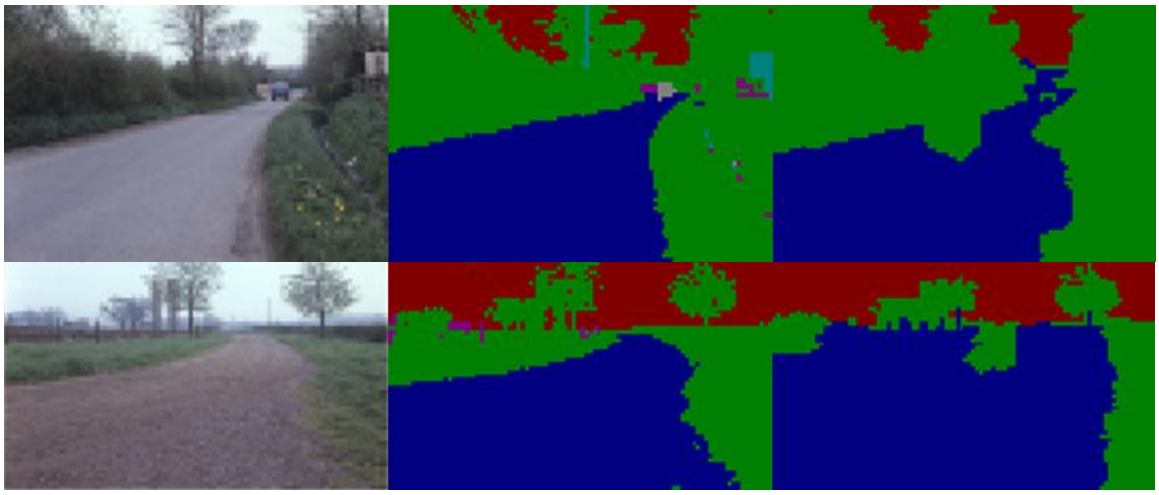

图 6.2 sowerby 预测效果图. 第一列为原图像，第二列为掩码图片，第三列为 linearsvm 预测结果.

结　论

通过理论分析与实验验证，基于超像素的多特征融合是一种有效的目标识别方法，能将多种不同属性同时考虑在内。尽管如此，但这并不代表永远都要采用多特征融合的方式，过分的使用也会导致负面的效果，这取决于数据集，实际中应该尝试多种可能的特征组合情况，尽量选择目标与背景差异较大的特征。相对于非训练的方法，本文的基于超像素的训练方式不仅数据依赖性低，而且因为超像素的引入使训练特征的数量比直接在像素点特征的基础上训练要减少很多，这提高了训过程练的效率。本文使用条件随机场对支持向量机的结果做进一步改善，提高了训练和预测结果的精度。

尽管在本文框架的基础上实验获得了一些结果，但其中仍然存在许多为解决的问题及需要改进的地方。首先受限于计算机的硬件，无法对大量的数据进行训练。本文尚未实现多类目标识别中的条件随机场优化支持向量机的过程，这需要进一步的对支持向量机的分类过程的理解。尽管使用了多种特征及超像素的方法，但这些方法的参数选择敏感性较高，而且提取超像素的参数与提取像素点特征的参数之间应该保持一定的相关性，才能让超像素中的统计点在合理的范围内，这些都需要进一步改善与提高。

参 考 文 献

[1] Koen E.A. and Van de Sande. Evaluating Color Descriptors for Object and Scene Recognition. Pattern Analysis and Machine Intellignece. IEEE. Sept, 2010.

[2] Brian Fulkerson and Andrea Vedaldi. Class Segmentation and Object Localization with Superpixel Neighborhoods. In Proc. ICCV, 2009.

[3] Cortes C, Vapnik V. Support vector machine[J]. Machine learning, 1995, 20(3): 273-297.

[4] Yunpeng Li, Noah Snavely, and Dan Huttenlocher. Location Recognition Using Prioritized Feature Matching. In Proc. ECCV, September 2010.

[5] J. Lafferty, A. McCallum, and F. Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In Proc. ICML, 2001.

[6] Sutton C, McCallum A. An introduction to conditional random fields for relational learning[J]. Introduction to statistical relational learning, 2007, 93: 142-146.

[7] Leibe B., Micolajckzyk K., Schiele, B. Efficient clustering and matching for object class recognition. In Proc. BMVC, 2006.

[8] Aurelien Lucchi, Yunpeng Li, Kevin Smith and Pascal Fua. Structured Image Segmentation using Kernelized Features. In Proc. ECCV, October 2012.

[9] Richard O.Duda and Perter E.Hart 等著. 李宏东等译. 模式分类（第二版）. 北京：机械工业出版社，2003，432-440，446-447.

[10] D. Lowe. Distinctive image features from scale-invariant keypoints. IJCV, 60(2): 91–110, November 2004.

[11] Varma and Zisserman. A statistical approach to texture classification from single images. ICJV: Special Issue on Texture Analysis and Synthesis, in 2005.

[12] Wei He, Takayoshi Yamashita, Hongtao Lu, and Shihong Lao. Surf Tracking. In Proc. ICCV, 2009.

[13] Chunhui Gu, Joseph J.Lim, Pablo Arbelaez and Jitendra Malik. Recognition using Regions. In Proc. CVPR, 2009.

[14] A. Vedaldi and S. Soatto. Quick shift and kernel methods for mode seeking. In Proc. ECCV, 2008.

[15]Radhakrishna Achanta, Appu Shaji, Kevin Smith, Aurelien Lucchi, Pascal Fua, and Sabine Süsstrunk. SLIC Superpixels. EPFL Technical Report no. 149300, June 2010.

[16]C. Elkan. Using the triangle inequality to accelerate k-means. In Proc. ICML, 2003.

[17]Svetlana Lazebnik, Cordelia Schmid, and Jean Ponce. Beyond Bags of Features: Spatial Pyramid Matching for Recognizing Natural Scene Categories. In Proc. CVPR, 2006.

[18]Brian Fulkerson and Andrea Vedaldi. Localizing Objects with Smart Dictionaries.In Proc.ECCV, 2008.

[19]Martin Szummer, Pushmeet Kohli, Derek Hoiem. Learning CRFs using Graph Cuts. In ECCV, 2008.

[20]C.-C. Chang and C.-J. Lin. LIBSVM: a library for support vector machines, 2001. Software available at http://www.csie.ntu.edu.tw/~cjlin/libsvm/.

[21]Rong-En Fan and Kai-Wei Chang. LIBLINEAR: A Library for Large Linear Classification. July 4, 2012. Software available at http://www.csie.ntu.edu.tw/~cjlin/liblinear/.

[22]Yuri Boykov, Olga Veksler, Ramin Zabih. Fast Approximate Energy Minimization via Graph Cuts. IEEE transactions on PAMI, vol. 20, no. 12, p. 1222-1239, November 2001.

[23]Yuri Boykov, Vladimir Kolmogorov. An Experimental Comparison of Min-Cut/Max-Flow Algorithms for Energy Minimization in Vision.

[24]P. F. Felzenszwalb and D. P. Huttenlocher. Efficient graph-based image segmentation. IJCV, 59(2), 2004.

[25]A. Vedaldi and B. Fulkerson. VLFeat: An open and portable library of computer vision algorithms. http://www.vlfeat.org , 2008.

[26]M.Marszalek and C.Schmid. Accurate object localization with shape masks. In Proc. CVPR, 2007.

附录 A：外文原文

Survey of contemporary trends in color image segmentation（PARTS）

Sreenath Rao Vantaram
Rochester Institute of Technology
Chester F. Carlson Center for Imaging Science
Rochester, New York 14623
E-mail: sreenath.rit@gmail.com
Eli Saber
Rochester Institute of Technology
Chester F. Carlson Center for Imaging Science
Rochester, New York 14623
and
Rochester Institute of Technology
Department of Electrical and Microelectronic Engineering
Rochester, New York 14623

Abstract. In recent years, the acquisition of image and video infor-mation for processing, analysis, understanding, and exploitation of the underlying content in various applications, ranging from remote sensing to biomedical imaging, has grown at an unprecedented rate. Analysis by human observers is quite laborious, tiresome, and time consuming, if not infeasible, given the large and continuously rising volume of data. Hence the need for systems capable of automatically and effectively analyzing the aforementioned imagery for a variety of uses that span the spectrum from homeland security to elderly care. In order to achieve the above, tools such as image segmentation provide the appropriate foundation for expediting and improving the effective-ness of subsequent high-level tasks by providing a condensed and pertinent representation of image information. We provide a compre-hensive survey of color image segmentation strategies adopted over the last decade, though notable contributions in the gray scale domain will also be discussed. Our taxonomy of segmentation techniques is sampled from a wide spectrum of spatially blind (or feature-based) approaches such as clustering and histogram thresholding as well as spatially guided (or spatial domain-based) methods such as region growing/splitting/merging, energy-driven parametric/geometric active contours, supervised/unsupervised graph cuts, and watersheds, to name a few. In addition, qualitative and quantitative results of promi-nent algorithms on several images from the Berkeley

segmentation dataset are shown in order to furnish a fair indication of the current quality of the state of the art. Finally, we provide a brief discussion on our current perspective of the field as well as its associated future trends. © 2012 SPIE and IS&T. [DOI: 10.1117/1.JEI.21.4.040901].

1 Introduction

Color image segmentation facilitates the separation of spatial-spectral attributes contained in images into their individual constituents; a task that is accomplished quite comfortably by our visual system and cortical mechanisms. However, mimicking this capability of human observers in an artificial environment has been found to be an extremely hallenging problem. Formally, color image segmentation is defined as the process of partitioning or segregating an image into regions (also called as clusters or groups), manifesting homogeneous or nearly homogeneous attributes such as color, texture, gradient as well as spatial attributes pertaining to location. Fundamentally, a segmentation algorithm for an image is said to be “complete” when it provides a unique region or label assignment for every pixel, such that all pixels in a segmented region satisfy certain criteria while the same principles are not universally satisfied for pixels from disjoint regions.

The cardinal motivation for image segmentation is two-fold. It not only provides an end user with the flexibility to efficiently access and manipulate individual content, but also furnishes a compact representation of the data wherein all subsequent processing can be done at a region/segment level as opposed to the pixel level, resulting in potentially significant computational gains. To this effect, segmentation is predominantly employed as a preprocessing step to anno-tate, enhance, analyze, classify, categorize, and/or abstract information from images. In general, there are many appli-cations for color image segmentation in the image proces-sing, computer vision, and pattern recognition fields, including content-based image retrieval (CBIR), image ren-dering, region classification, segment-based compression,surveillance, perceptual ranking of regions, graphics, and multimedia to name a few. Furthermore, many approaches have been developed in other modalities of imaging such as remote sensing (multi/hyperspectral data) and biomedical imaging [computed tomography (CT)], positron emission tomography (PET), and magnetic resonance imaging (MRI) data for sophisticated applications such as large area

search,three-dimensional (3-D) modeling, visualization, and navi-gation. The exponential growth of the number of applica-tions that employ segmentation in itself provides a strong motivation for continued research and development.

In the context of color imagery, segmentation is often viewed as an ill-defined problem with no perfect solution but multiple generally acceptable solutions due to its sub-jective nature. The subjectivity of segmentation has been extensively substantiated in experiments1 conducted at the University of California at Berkeley to develop an evaluation benchmark, where a database of manually generated seg-mentations of images with natural content was developed using multiple human observers. In Fig. A.1(a), 10 images (arbitrarily named airplane, starfish, race cars, hills, boat,church, cheetah, dolphins, lake, and skydiver) from the aforementioned database are displayed. Additionally, several manually segmented ground truths with region boundaries superimposed (in green) on the original image are shown in Fig. A.1(b) to A.1(f). Analysis of the obtained ground truth results by Martin et al. divulged two imperative aspects: 1. an arbitrary image may have a unique suitable segmenta-tion outcome while others possess multiple acceptable solu-tions, and 2. the variability in adequate solutions is primarily due to the differences in the level of attention (or granularity) and the degree of detail from one human observer to another,as seen in Fig. A.1. Consequently, most present day algorithms for segmentation aim to provide generally acceptable out-comes rather than a “gold standard” solution.

There are several excellent surveys of image segmentation strategies and practices. The studies done by Fu et al.2 and Pal et al.3 are amongst the earliest ones that have been widely popular. In their work, Fu et al.2 categorized segmentation approaches developed during the 1970s and early 1980s for gray scale images into three classes; namely, clustering,edge detection, and region extraction. On the other hand,Pal et al.3 reviewed more complex segmentation techniques established in the late 1980s and early 1990s that involved fuzzy/nonfuzzy mechanisms, markov random fields (MRFs) probabilistic models, color information as well as neural net-works—all of which were still in their early stages of devel-opment. The surveys done by Lucchese et al.4 and Cheng et al.5 were among the first that exclusively provided an in-depth overview of algorithms targeted at segmenting color images, instituted throughout the 1990s.

In this paper, we provide a comprehensive overview of the image segmentation realm

with the goals to: 1. facilitate access to contemporary procedures and practices developed in the recent past (2001 to current), 2. establish current standards of segmentation outcomes from a qualitative and quantitative standpoint by displaying results acquired from state-of-the-art techniques, 3. discuss our view on the field as it stands today, and 4. outline avenues of future research. The remainder of this paper is organized as follows. Section 2 provides broad and specific categorizations of segmentation approaches based on their inherent methodology and illus-trates experimental results derived from prominent color image segmentation algorithms. Furthermore, Sec. 3 pro-vides a brief quantitative evaluation of the aforementioned algorithms. Finally, conclusions and future directives are presented in Sec. 4.

2 Classification of Segmentation Methodologies

Segmentation procedures can be broadly categorized from a high level perspective as well as specifically grouped based on their technical grounding (low level classification). The following subsections describe each of the two taxonomies in detail.

2.1 High-Level Taxonomy

Image segmentation techniques can, in general, be broadly classified (see Fig. A.2) based on: 1. the image type, 2. the extent of human interaction, 3. the manner in which the image is represented for processing, 4. the number and type of attributes used and, 5. the fundamental principle of operation.

The first criterion segregates algorithms that have been developed for monochrome (or single band) images from the ones that handle color (or three band) images. The second criterion distinguishes methods that require human interven-tion (supervised processes) for segmentation from the ones that operate without any manual interference (fully automatic or unsupervised processes). The third criterion separates seg-mentation procedures that directly operate on the original image (single scale configuration) from the ones that operate on multiple representations of the image (multiscale config-uration), each manifesting different amount of information.

The fourth criterion differentiates algorithms based on the type of image information (e.g., gray/color intensity, gradi-ent/edge, or textural features) utilized to perform the segmen-

tation. It is imperative to note that most methods use the aforementioned image attributes individually (single attri-bute methods) or in specific combinations (multiple attribute methods) to categorize them. Finally, the last criterion based on the underlying principle of operation discriminates seg-mentation mechanisms as being either spatially blind or spatially guided. Spatially blind approaches as the name sug-gests are techniques that are "blind" to spatial information, or, in other words, do not take into account the spatial arrangement of pixels in an image. Instead these methods rely heavily on grouping image information in a suitable attribute/feature space. On the other hand, spatially guided approaches tend to exploit the spatial arrangement of pixels in an image during the segmentation process.

2.2 Low-Level Taxonomy

As mentioned previously, most segmentation modus oper-andi can be viewed as being either spatially blind or spatially guided. It is this distinction that forms the basis of our low-level taxonomy where we specifically group segmentation procedures based on their technical components, as depicted in Fig. A.3.

2.2.1 Spatially blind approaches

Spatially blind approaches perform segmentation in certain attribute/feature spaces, predominantly related to intensity (gray or color). Popular segmentation techniques that fall within the notion of being spatially blind involve clustering and histogram thresholding.

Clustering. In its simplest form, clustering is a spatially blind technique wherein the image data is viewed as a point cloud on a one-dimensional (1-D) gray scale axis or in a 3-D color space (see Fig. A.4) depending on the image type.

Several different color spaces—such as RGB, Commis-sion International de l'Eclairage (CIE) L*a*b* and L*u*v*, YCbCr, HSI etc., to name a few—with different properties have been extensively utilized for segmentation.6 The essence of a typical clustering protocol is to analyze this gray/color intensity point cloud and partition it using pre-defined metrics/objective functions to identify meaningful pixel groupings also known as classes or clusters. Further-more, the clustering process is done such that, when com-plete, the pixel data within a specific class possess, in general, a high degree of similarity while the data between classes has low affinity. The biggest advantage of clustering approaches over others is inherent in their simplicity and ease of implementation. However, since the point

clouds generated are image dependent, selecting/initializing the number of clusters so as to obtain semantic partitioning with respect to the image being processed is a challenging task, especially in the case of color imagery. Furthermore, as the dimensionality of the feature space is increased exponen-tially, acquiring definitive clusters becomes an arduous task.

Although many clustering approaches have been devel-oped for various applications, partitioning a feature space using a specific set of fixed points is the most widely adopted approach. Voronoi tessellation (VT) is a procedure in which a feature space is decomposed into various clusters (called Voronoi cells/regions) using a fixed set of points called sites, such that each observation in the feature space is assigned to the closest site predicated on a certain distance metric. More specifically, if X is a feature space constrained with a distance function d, and ðPk Þk ∈ K is a set of K sites in the space, then a Voronoi cell V k formed using the site Pk is the set of all points x ∈ X that satisfy:

$$V_k = [x \in X \mid d(x, P_k) \le d(x, P_j) \forall j \ne k] \qquad \text{(公式 A.1)}$$

where $d(x, P_k)$ represents the distance from x to Pk. Arbel áez et al.7 proposed a VT-based image segmentation technique utilizing color and lightness information derived from the image. The segmentation objective was achieved in a two-step process comprised of: 1. presegmentation and 2. hierarchical representation. The presegmentation step employed a VT process wherein the extreme components of the lightness (LÃ) channel were used as sites to form an extrema mosaic of Voronoi regions. The second step involved the development of a stratified hierarchy of parti-tions derived from the extrema mosaic using a pseudo-distance metric called ultrametric, specifically defined for color images. Subsequently, a single real-valued soft bound-ary image called the ultra-metric contour map (UCM) was constructed to arrive at the final segmentation.

Centroidal voronoi tessellation (CVT) is a special cate-gory of VT wherein the sites producing Voronoi cells are chosen equivalent to their center of mass. Wang et al.8 gen-eralized the basic CVT by integrating an edge-related energy function with a classic clustering energy metric to propose an edge-weighted centroidal voronoi tessellation (EWCVT) for effective segmentation of color images. CVTs form the core of many prominent clustering algorithms such as K-means. The K-means algorithm partitions a set

of n-pixels into K clusters by minimizing an objective function. From a color segmentation perspective, the aforementioned algorithm analyzes the image data in a 3-D space to eventually identify K-sites (known as cluster centers or means) such that the mean squared distance from each data point to its nearest center is minimized. To this effect, in an arbitrary iteration (called as a Voronoi iteration or Lloyd's algorithm), the entire color space is separated into K partitions by assigning each observation to the cluster with the closest center (note initialization in the first iteration may be randomly done). Following this, a new estimate of the cluster center is computed based on the current cluster assignment information and is utilized as an input to the next iteration of the algorithm. The algorithmic steps described above are repeated until convergence is achieved. McQueen9 was the first to employ the K-means algorithm to handle multivariate data. Among recent advances, Kanungo et al.10 proposed an efficient version of the algorithm called the "filtering algorithm," by utilizing a k-dimensional (kd) tree representation of the image data. For each node of this tree, a set of candidate centers were determined and strategically filtered as they were propagated to the node's children. The biggest advantage of this approach was that, since the kd-tree representation was formed from the original data rather than from the computed centers, it did not mandate an update in its structure for all iterations, in contrast to the conventional K-means architecture. Chen et al.11 employed a generalization of the classical K-means algorithm better known as the adaptive clustering algorithm (ACA), with spatially adaptive features pertaining to color and texture, to yield perceptually tuned segmentations. Consequently, the ACA method is an exception to the norm of spatially blind clustering. In his work, Mignotte12 proposed a novel color image segmentation procedure based on the fusion of multiple K-means clustering results by minimizing the Euclidean distance function, obtained from an image described in six different color spaces namely RGB, HSV, YIQ, XYZ, LAB, and LUV. Once the label fields from each of these color spaces are obtained, a local histogram of the class labels across the aforementioned label fields is computed for each pixel, and the set of all histograms are employed as input feature vectors to a fusion mechanism that culminates in the final segmentation output. The fusion scheme is comprised of the K-means algorithm using the Bhattacharya similarity coefficient, which is a histogram-based similarity metric. The algorithm in Mignotte12 was further enhanced in Mignotte13 by using a spatially

constrained K-means labeling process in place of the fusion mechanism to arrive at the optimal result. While the prior algorithm developed by Mignotte was aimed at exploring the possibility of integrating multiple segmentation maps from simple data partitioning models to obtain an accurate result, the later algorithm was novel in the sense that within the K-means framework implicit spatial associations in an image were taken into account (similar to the work in Ref. 14) to uncover the best solution, and consequently the process was not completely spatially blind.

Mean shift clustering15 is another routine that has had pervasive use for gray/color image segmentation within the last decade. This generic nonparametric technique facilitates the analysis of multidimensional feature spaces with arbitrarily shaped clusters, based on the "mean shift" concept, originally proposed by Fununaga et al.16 The mean shift procedure is a kernel density estimation (or Parzen window-based technique) that scrutinizes a feature space as an empirical probability density function (pdf) and considers the set of pixel values from an arbitrary image as discrete samples of that function. The procedure exploits the fact that clusters/dense regions in a feature space typically manifest themselves as modes of the aforementioned pdf. In what follows, if S is a finite point cloud in an n-dimensional Euclidean space, X and K is a symmetric Kernel function of specific characteristics, then the sample mean m ðxÞ at a pixel x ∈ X computed utilizing a weighted combination of its nearby points determined by K is given as:

$$m(x) = \frac{\sum_{s \in S} K(s-x)s}{\sum_{s \in S} K(s-x)} \qquad \text{（A.2）}$$

To this effect, at every pixel location x, a mean shift vector $m(x)-x$ is obtained with K centered at x, such that the vector points towards the direction of the maximum increase in density. Subsequently, the operation $x \leftarrow m(x)$ is performed that shifts the value of x toward the mean followed by the re-estimation of $m(x)$. This process is repeated until convergence of $m(x)$ is achieved. At the end of the mean shift process, the closest peak in the pdf is identified for each pixel. Since the mean shift algorithm uses spatial knowledge in its framework, it also represents an exception to conventional spatially blind clustering. Mean shift clustering guided by edge information was first seen in the work by

Christoudias et al.,18 who proposed the edge detection and image segmentation (EDISON) system, aimed at improving the sensitivity of extracting homogeneous regions while maintaining or ideally minimizing over-segmentation of an image. Figure A.5 illustrates a few results of the EDISON system using default parameters (spatial band hs ¼ 7, color band width hr ¼ 6.5, and minimum region size M ¼ 20). Hong et al.19 proposed an improved version of the mean shift segmentation algorithm by incorporating: 1. an enhanced technique for mode detection, 2. an optimized process for the global analysis of the locally identified modes, and 3. the elimination of textured areas in order to achieve stable results in various background conditions. Ozden et al.20 pioneered an effective technique that combined low-level color and spatial and texture features in the mean shift framework for color image segmentation.

Neural networks-based data clustering is a category that has originated exclusively from the field of artificial intelligence. Within this domain, methods involving selforganizing maps (SOMs) have received the most attention in the last decade. A self-organizing map or a self-organizing feature map (SOFM)—alternately known as a Kohonen map—is a specific kind of artificial neural network (ANN) that was first introduced by Kohonen21 as a tool for providing intelligent representations of high/multi-dimensional feature spaces in significantly lower (one or two) dimensions. A SOM (shown in Fig. A.6) comprises of an input layer of nodes/neurons organized in a vector whose size is equivalent to the dimensions of the input feature space. Each node is connected in parallel to a two-dimensional (2-D) output layer of neurons in a rectangular or hexagonal arrangement as well as their corresponding neighboring neurons utilizing an appropriate weighting scheme that signifies the strength of various connections. A SOM operates in a “training” phase that gradually constructs a feature map using a subset of samples from the input feature space, followed by a mapping routine in which an arbitrary new input sample is automatically classified. At the culmination of the two modes of operation, a low-dimensional map that reflects the topological relationships of samples in the feature space predicated on their similarity is subsequently generated. In other words, samples that have similar characteristics in the input feature space form distinct clusters in this map.

Huang et al.22 developed a color image segmentation methodology that employed a two-stage SOM-based ANN. The algorithm is initiated by an RGB to HVC (hue-value-

chroma) color conversion of the input image, which is employed by an SOM to identify a large initial set of color classes. The resultant set of classes are further refined by first computing the normalized Euclidean distance among them, and the obtained between-class distances are furnished as inputs to a second SOM that identifies the final batch of segmented clusters. In a similar scheme, Ong et al.23 constructed a color image segmentation technique based on a hierarchical two-stage SOM in which the first stage identifies dominant colors in the input image presented in the L*A*B* color space, while the second stage integrates a variablesized 1-D feature map and cluster merging/discarding operations to acquire the eventual segmentation result. Li et al.24 demonstrated an effective color image segmentation approach using a SOM and the fuzzy-C-means (FCM) clustering procedure. The algorithm is initiated by finding wellsuited image-dependent features derived from five different color spaces using a SOM. Subsequently, the obtained features were employed in a FCM protocol to attain the output segmentation. Dong et al.25 instituted two alternate ANN based strategies for color image segmentation. The first strategy was unsupervised. It involved distinguishing a set of color prototypes using SOM-based learning from the input image represented in the LÃuÃvÃ color space. These color prototypes were passed on to a simulated annealing-driven clustering routine to yield well-defined clusters. The second method, built off the aforementioned algorithm, was coupled with hierarchical pixel learning (that generated different sizes of color prototypes in the scene) and classification protocols to provide more accurate segmentation outcomes in a supervised fashion. Partitioning of color imagery using SOM and adaptive resonance theory (ART) was first seen in the work of Yeo et al.,26 who proposed two compound ANN architectures called SOMART and SmART (SOM unified with a variation of ART) that yielded improved segmentations in comparison to traditional SOM based techniques. On the other hand, Araújo et al.27 designed a fast and robust self-adaptive topology ANN model called local adaptive receptive field SOM (LARFSOM) that deduced compact clusters and inferred their appropriate number based on color distributions learned rapidly from the network's training phase using a small percentage of pixels from the input image. The algorithm was tested on color images with varying segmentation complexities and was found to outperform several prior SOM-based techniques. Frisch28 introduced a novel approach robust to illumination variations that employed SOMs for the construction of

fuzzy measures applicable to color image segmentation. This work was a unique attempt wherein efficient fuzzy measures, to accomplish the segmentation task, were derived using SOM-based processing. Ilea et al.29 devised a fully automatic image segmentation algorithm called CTex using color and texture descriptors. The CTex segmentation architecture first extracts dominant colors from the input image presented in the RGB and YIQ color spaces using an SOM classifier. In doing so, the appropriate number of clusters (K) in the scene are also identified. Subsequently, a conventional K-means clustering algorithm is employed in a six-dimensional (6-D) multispace spanned by both the above stated color spaces to obtain a segmentation result purely based on color information. This is followed by the computation of textural features using a Gabor filter bank, which, along with the previously acquired segments, are provided as input to a novel adaptive spatial K-means (ASKM) clustering algorithm that delineates coherent regions of color and texture in the input image.

The clustering techniques discussed so far are typically categorized as hard clustering approaches since every observation in the feature space has a unique and mandatory cluster assignment yielding clusters with sharp boundaries. In contrast, significant work has been done for the advancement of fuzzy clustering methods that facilitate observations to bear a certain degree of belongingness or membership to more than one cluster, resulting in overlapping clusters and/or clusters with “soft” boundaries. The Fuzzy-C-Means (FCM) algorithm, originally developed by Dunn30 is the most widely utilized fuzzy clustering methodology and is similar to the K-means technique, partitions a set of n-pixels ($X = \{x_1 ..., x_n\}$) into C fuzzy clusters ($C = \{c_1 ..., c_n\}$) by minimizing an objective function. The objective function utilized by the FCM algorithm is represented as:

$$J_m = \sum_{i=1}^{n} \sum_{j=1}^{c} u_{ij}^m \left\| x_i - c_j \right\|^2 \qquad \text{（A.3）}$$

where

$$u_{ij}^m = \frac{1}{\sum_{k=1}^{c} \left(\frac{\left\| x_i - c_j \right\|}{\left\| x_i - c_k \right\|} \right)^{2/(m-1)}} \quad \text{, and } \quad c_j = \frac{\sum_{i=1}^{n} u_{ij}^m x_i}{\sum_{i=1}^{n} u_{ij}^m} \qquad \text{（A.4）}$$

From Eqs. (3) and (4) it can be inferred that the FCM objective function differs from K-

means by incorporating membership values uij for various observations xi in the feature space as well as a “fuzzifier” term $m \mid m \in \{1 \le m \le \infty\}$ that directs the extent of cluster fuzziness.

In their work, Yang et al.31 proposed two eigen-based fuzzy clustering routines namely, separate eigenspace FCM (SEFCM) and couple eigen-based FCM (CEFCM), for segmentation of objects with desired attributes in color imagery. Given an arbitrary image with a predefined set of pixels, the color space in which the image is expressed is initially divided into two eigenspaces called principal and residual eigenspaces using the Principal Component Transformation. Following this, the SEFCM design obtains a segmentation output by integrating the results of independently applying the FCM algorithm to the aforementioned eigenspaces. The integration process involves a logical selection of common pixels from the two clustering results. On the other hand, the CEFCM arrangement obtains an output segmentation result by jointly considering the principal and residual eigenspaces. Both routines were found to outperform the
traditional FCM clustering approach from a color object segmentation perspective. Liew et al.32 instituted an adaptive fuzzy clustering scheme by imposing local spatial continuity using contextual information. The method was targeted for exploiting inter-pixel correlation existent in most conventional imagery in a fuzzy framework. Chen et al.14 proposed a computationally efficient version of the FCM algorithm using a two-phase scheme involving data reduction followed by clustering. This computationally more efficient approach was found to produce results of similar quality to the conventional FCM. More recently, Hung et al.33 developed a weighted FCM (WFCM) clustering technique wherein the weights for various features were computed using a bootstrap method. Incorporating the bootstrap approach was found to provide satisfactory weights to individual features from a statistical variation viewpoint and enhance the performance of the WFCM procedure. Tziakos et al.34 proposed an approach using the Laplacian Eigen (LE) map algorithm, a manifold learning technique, to boost the performance of FCM clustering. The methodology is commenced by extracting local image characteristics from overlapping regions in a high dimensional feature space, from which a low-dimensional manifold was learned using spectral graph theory. Following this, the LE-based dimensionality reduction technique was used to compute a low dimensional map that

captured local image characteristic variations, eventually used to enhance the performance of FCM clustering. Krinidis et al.35 and
Wang et al.36 developed alternate yet efficient versions of the FCM scheme that employed both local intensity and spatial information. Yu et al.37 founded an ant colony-fuzzy C-means hybrid algorithm (AFHA) for color image segmentation that adaptively clustered image elements utilizing intelligent cluster center initializations as well as subsampling for computational efficiency. The results of the AFHA structure were found to have smaller distortions and more stable cluster centroids over the conventional FCM.

Besides the practices discussed so far in this section, several unique clustering-based methods for image segmentation have also been proposed. Veenman et al.38 developed an efficient and optimized model for clustering using a cellular co-evolutionary algorithm (CCA) that does not require any prior knowledge of the number of clusters. On the other hand, Allili et al.39 instituted a clustering model that combined a generalized Gaussian mixture model with a pertinent feature selection to alleviate problems of under/over segmentation. Jeon et al.40 introduced a sparse clustering method for color image data using the principle of positive tensor factorization (PTF). Aghbari et al.41 proposed a hillmanipulation process where the protocol of segmenting an arbitrary color image was treated in an equivalent fashion to that of finding hills in its corresponding 3-D intensity histogram. Ma et al.42 introduced the notion of clustering using lossy data coding and compression and demonstrated their work on natural scene color images. The algorithm in Ma et al.42 was employed by Yang et al.,43 who proposed a compression-based texture merging (CTM) routine that treated segmentation as a task of clustering textural features modeled as a mixture of Gaussian distributions, wherein the clustering methodology was acquired from a lossy data compression protocol. Sample segmentation outcomes of the CTM algorithm using default parameters (coding data length parameter γ ¼ 0.2) are exhibited in Fig. A.7. Huang et al.44 advocated the concept of pure “clustering-then-labeling” for efficient segmentation of color images.

Histogram thresholding. Histogram thresholding [see Ref. 45 for a comprehensive survey] is a spatially blind technique primarily based on the principle that segments of an image can be identified by delineating peaks, valleys, and/or shapes in its corresponding intensity histogram. Similar to clustering, histogram thresholding protocols require minimal effort to

realize in comparison with most other segmentation algorithms and function without the need for any a priori information about the image being partitioned. Owed to its simplicity, intensity histogram thresholding initially gained popularity for segmenting gray-scale images. However, during its course of development, it was found that thresholding intensity histograms did not work well for low-contrast images without obvious peaks and yielded ambiguous partitions in the presence of spurious peaks manifested by noise. Additionally, for color images, it was determined that thresholding in a multidimensional space is a difficult task. Figure A.8 illustrates color histograms of the starfish and boat images in the RGB space, generated using an open-source ImageJ plugin called Color Inspector 3D. Each color bin in the 3-D histogram is represented as a sphere whose volume is proportional to the frequency of occurrence of that color. From the histograms, it can be observed that finding multiple thresholds to efficiently partition them presents a challenging problem.

Kurugollu et al.47 proposed an algorithm for color image segmentation that involved two major steps, namely multithresholding and fusion. The method is initiated by forming 2-D histograms using pair-wise band combinations (RG,GB, and BR), each of which were subjected to a peak finding protocol. Following this, based on the delineated peaks, a multithresholding scheme was used to form a segmentation result unique to each pair of channels. These three segmentation results were fused using a label concordance algorithm and refined using a spatial chromatic majority filter to derive the final segmentation result. In a similar framework, Cheng et al.,48 designed a color image segmentation scheme, based on the idea of thresholding a homogram, which simultaneously captured the occurrence of gray levels along with adjoining homogeneity values among pixels. The segmentation routine was initiated by forming a homogram individually for each color channel, the peaks of which were used to guide a subsequent thresholding scheme to acquire an initial oversegmented set of clusters. Finally, the three sets of clustering results from the red, green, and blue planes, respectively, were united together to achieve the final segmentation. Mushrif et al.49 exploited the concept of Histon thresholding based on rough set theory to devise an efficient algorithm for color image segmentation. A Histon is defined as a set of pixels that are all potentially part of a particular segment. Their three-step architecture involved computing a Histon, followed by thresholding and culminating

in a region merging process (discussed in Sec. 2.2.2.1). Additionally, they further enhanced the aforementioned methodology though the work in Mushrif et al.50 using an Atanassov's intuitionistic fuzzy set (A-IFS) Histon representation of the input image. In their work, Manay et al.51 proposed an adaptive thresholding structure for fast segmentation using an anisotropic diffusion model based on the principle that, for an arbitrary local area, an optimal threshold can be derived close to image edges using a smooth version of it.

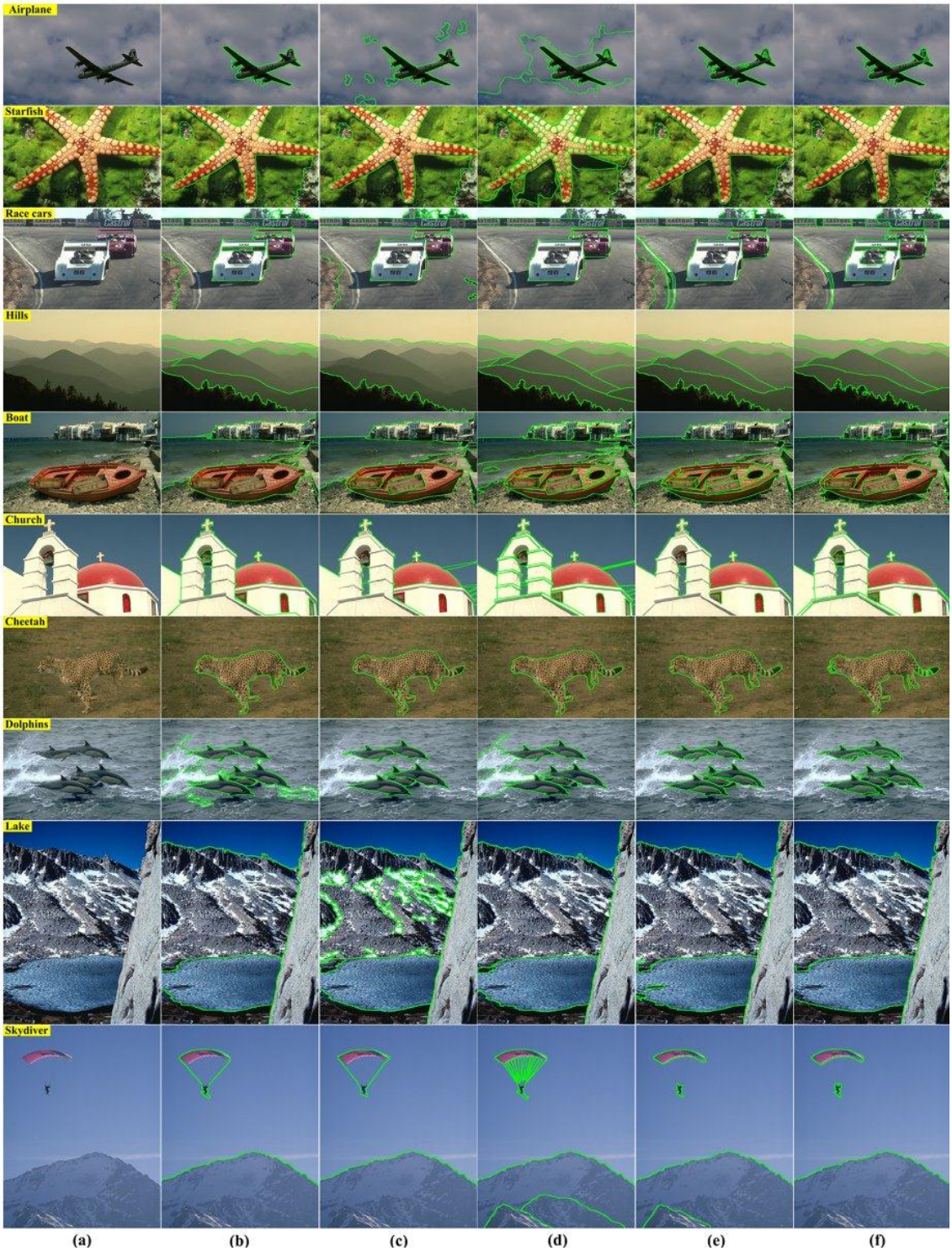

Fig. A.1 Berkeley segmentation benchmark: (a) original images, and (b) to (f) region boundaries of multiple manually generated segmentations overlaid on the images.

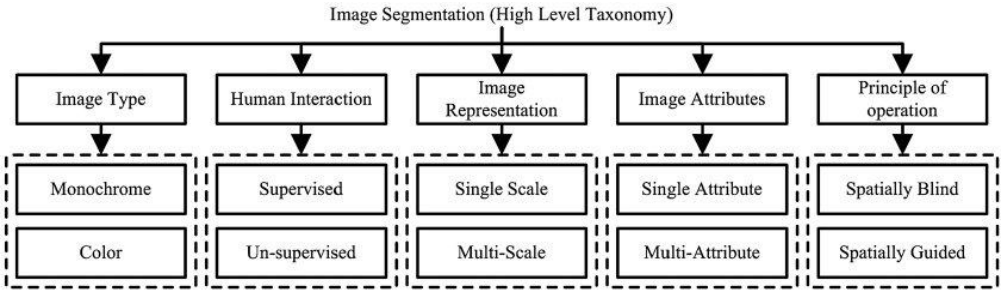

Fig. A.2 High-level taxonomy of image segmentation algorithms.

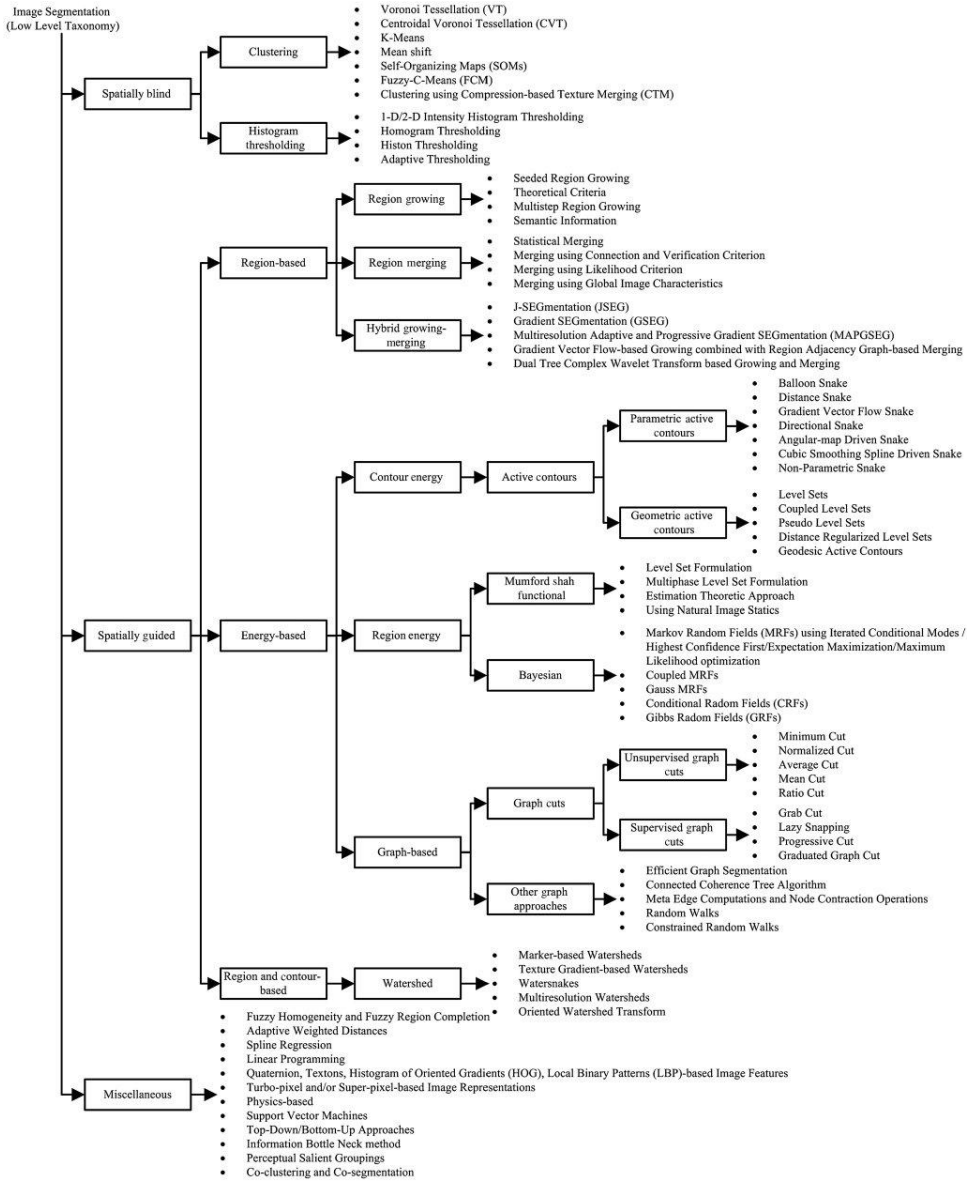

Fig. A.3 Low-level taxonomy of image segmentation algorithms.

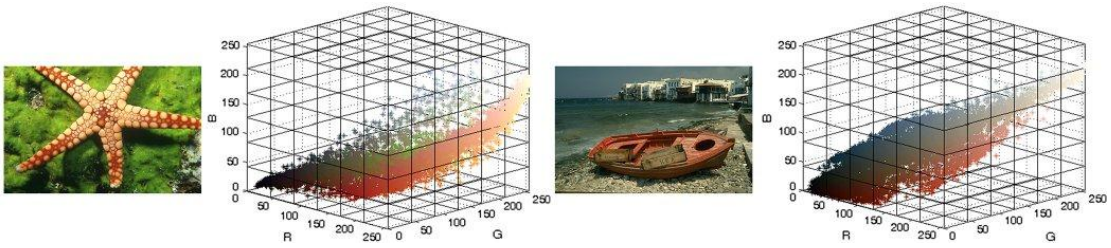

Fig. A.4 Sample color images with their corresponding 3-D point clouds.

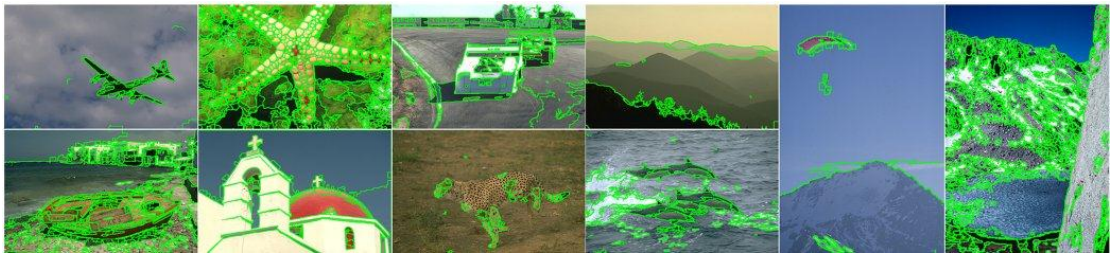

Fig. A.5 Results of the EDISON system.

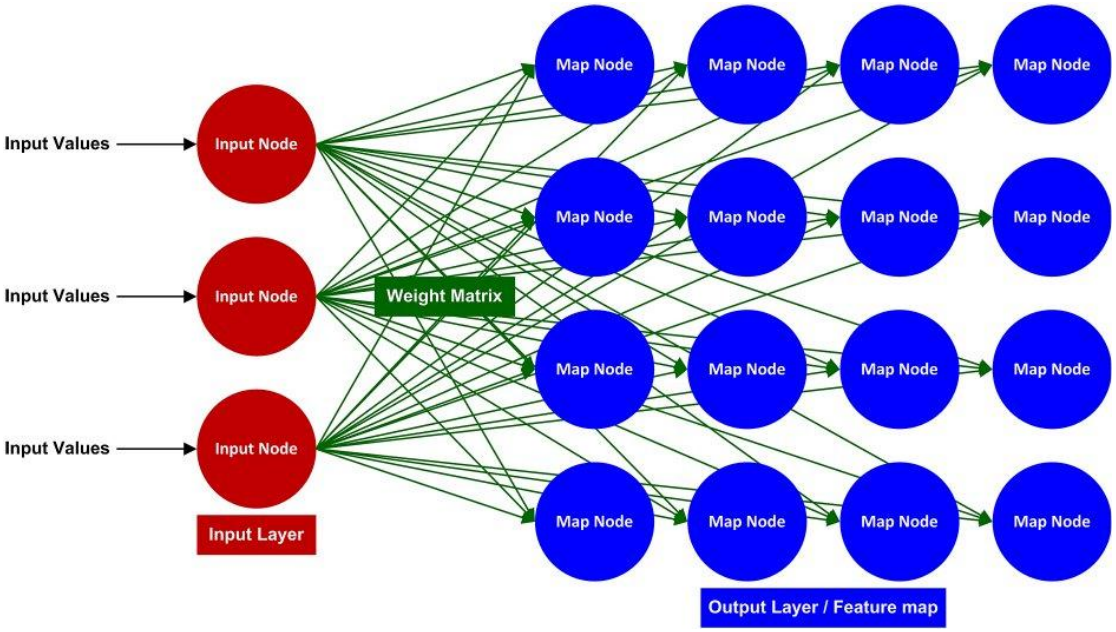

Fig. A.6 Self-organizing map (SOM) in a rectangular neural layout.

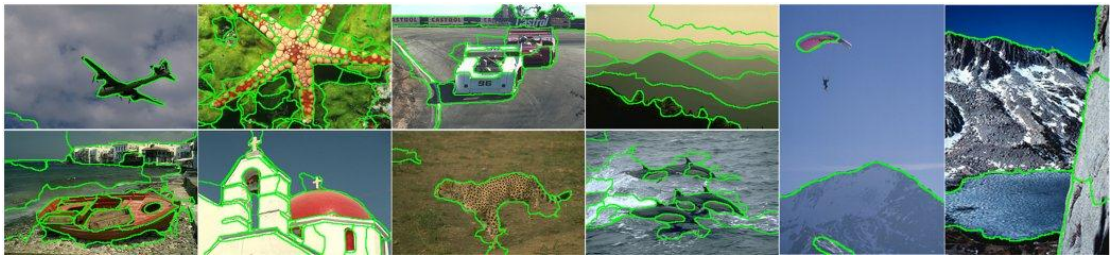

Fig. A.7 Results of the CTM algorithm.

附录 B：外文译文

彩色图像分割的现代趋势调查（部分）

（作者参见英文原文）

摘要：近年来，在各种各样的应用中，对图像和视频的采集信息进行处理、分析、理解和底层内容的利用，从远程传感到生物医学成像，都以前所未有的速度增长。如果可行的话，考虑到大量的连续的数据量增长，使用人力去分析这些东西非常费劲、无聊而且耗时。因此需要自动化的有效的系统去分析前面提到的各种应用图像，从国家安全到老年看护。为了实现上述目标，图像分割等工具提供了适当的基础，能通过提供一个浓缩的确切的图像信息表达方式，而促进与提高后续高级任务。我们对过去十年中使用的彩色图像分割策略进行了全面的调查，尽管在灰度域的显著贡献也将被讨论。我们的分割技术的分类方法是从一个大范围的空间谱（或基于特征）采样得到，方法如聚类和直方图阈值以及作为空间引导（或空间域的）方法如区域增长/分裂/合并，能源驱动参数/几何活动轮廓、监督/无人监督的图割，和分水岭等。此外，对伯克利的一些图像分割数据集的的著名算法的定性和定量结果将被展示，为了提供一个对当前最前沿技术的公平的展示。最终，我们提供了一个简单的讨论，关于当前领域的展望以及与之相关的将来趋势。

1 简介

彩色图像分割促进包含在图像到他们个别成分的空间谱属性的分离；一个容易完成的任务，通过我们的视觉系统和舒适的皮质机制。然而，模拟人类观察人造环境被认为是一个极具挑战的问题。正式的说，彩色图像分割定义成：通过操纵同类或近似相同属性，比如颜色、纹理、梯度以及与位置有关的空间属性，分割或分开一幅图像成多个区域（也称为类或组）。从根本上说，对于一幅图片，当它为每个像素提供了唯一的区域或标签分配时，认为一个分割算法结束。因此，分割区域中的所有像素满足一些标准，然而相同的原则对非连接的区域不通用。

图像分割的基本动机是双重的。不仅提供了一个灵活性的最终用户，高效地访

问和操纵单独的内容，而且提供了一个紧凑的数据表示，所有的后续处理可以在区域/分割级别，与像素级别相反，导致潜在的重要的计算增益。这个效应，分割主要用作图像注释的一个预处理步骤，更进一步，则是图像的分析，分类以及/或者摘要信息。通常的，在图像处理、计算机视觉和模式识别领域有许多彩色图像分割的应用，包括基于内容的图像检索（CBIR）、图像渲染、区域分类、基于分割的压缩、监视、感性的区域排名、图形以及多媒体等等。此外，许多方面已经以图像的其他形式开发，比如遥感（多/高光谱数据）和生物医学成像（CT），正电子发射（PET）、断层扫描和磁共振成像（MRI），对于复杂的应用比如大面积搜索，三维（3D）建模，可视化和导航。采用分割的应用的数量呈指数增长，其本身提供了一个强大的进一步的研究和发展的动力。

本文中的彩色图像，分割通常被认为是一个没明确定义没完美解决方案的问题，但有多个普遍接受的由于主观性质而可接受的方案。图像分割的主观性广泛的反映在加州大学伯克利分校的实验上，为了开发一个评价基准，通过多人观察，手动分割具有自然内容的图像的数据库。在图 1(a)中，显示了从前面提到的数据库中的 10 幅图片（任意命名的飞机、海星、赛车、山丘、船、教堂、猎豹、海豚、湖泊和降落伞）。除此之外，图 1(b)到图 1(f)中显示了一些原始图像在手动分割时区域边界重叠（绿色）的基准事实。通过分析 Martin 等人获得的基础事实结果，透露出两个重要的方面：1. 一个任意的图像可能有一个独特的合适的分割结果而其他人拥有多个可接受的解决方案，2. 恰当解决方案的变化主要由于不同程度的关注（或粒度）以及从一个观察者到另一个观察者的细节程度，就像图 1 看到的。因此，现在的大多数分割算法，目的在于提供通常可接受的结果，而不是一个“黄金标准”的解决方案。

有一些优秀的关于图像分割策略和实践的调查。由 Fu 和 Pal 等人的研究是最早的那些被广泛接受的。在它们的工作中，Fu 等人分类的分割方法在 1970 年代和 1980 年代早期，灰度图像分为三种类型；也就是，聚类、边缘检测以及区域提取。从另一方面，Pal 等人评估了更复杂的图像分割技术，这些技术建立在迟些的 1980 年代和 1990 年代，包括模糊/非模糊机制，马尔可夫随机场（MRFs）概率模型，颜

色信息以及神经网络——所有这些都出现在他们早期的开发中。Lucchese 和 Cheng 等人做的调查，针对分割彩色图像，第一个只提供了一个深入的概述的算法，在整个 1990 年代被使用。

在本文中，我们提供一个图像分割领域全面的概述，目标是：1. 促进同代人在最近的过去的发展上前进和实践，2. 从定性与定量的角度，通过展示从最前沿的技术中获取结果，建立当代的用于图像分割结果的标准，3. 讨论我们关于这个领域的观点，就像它今天展现的一样，4. 给出未来研究方向的轮廓。本文其余部分组织如下。第二节提供广泛而具体的图像分割方法的细分分类，这些方法基于它们固有的方法，并说明突出的彩色图像分割算法的实验结果。此外，第 3 节提供了关于前面提到的算法的简答评估。最后，第 4 节总结并且展示将来的发展方向。

2 分割方法的分类

分割过程可以大致分类，从高级别的角度以及基于技术基础的具体分组（低级别分类）。下面的子部分描述在细节上描述了两个分类方法。

2.1 高级别的分类方法

图像分割技术大致可分为（如图 B.2）基于：1. 图像类型，2. 人类交互的程度，3. 图像处理所展现的方式，4. 使用的属性的数量和类型，5. 操作的基本原则。

第一个标准，分割算法被开发用于单色图像（或单通道），还是彩色图像（或者三通道图像）。第二个准则，是需要在分割时用户交互（监督处理），还是无手工的操作（完全自动或者无监督处理）。第三个标准，是直接对原图像操作（单尺度配置），还是多表示方式的图像（多尺度配置），每个表示方式操作不同的大量的信息。第四个标准，基于用于执行分割的图像信息类型（比如，灰度/彩色强度，梯度/边缘，或者纹理特征）。必须注意，大多数方法使用前面提到的个别的图像属性（单属性方法）或在特定的组合（多属性方法），从而将它们分类。最后一个标准基于操作分割机制的基本原则，要么是空间盲目要么空间引导。空间盲目方法，如其名，对空间信息“盲目”，或者，换句话说，不考虑图像像素的空间分布。与这

些方法相反，则在一个合适的属性/特征空间严重依赖分组图像信息。另一方面，空间引导的方法趋向于在分割过程中使用图像像素的空间布局信息。

2.2 低级别的分类方法

就像前面提到的，大多数细分做法可以看作是空间盲目或空间引导。正是这种区别形成我们低级分类的研究基础，我们基于它们的技术组件细化组分割的过程，就像图 B.3 描述的那样。

2.2.1 空间盲目的方法

空间盲目方法在特定的属性/特征空间执行分割，与强度（灰度或颜色）关系密切。空间盲目的分割方法中，流行的分割技术包括聚类和直方图阈值。

聚类技术 最简单的格式，聚类是一种空间盲目的技术，在其中图像数据被看做在一维（1-D）灰度尺度坐标或一个依赖于图像类型的 3-D 彩色空间（如图 B.4）的一个点云。

一些不同的彩色空间——比如 RGB、Lab、Luv、YcbCr 以及 HIS 等，还有些其它的属性用于分割。有一个典型的聚类协议，用于分析这些灰度/彩色强度点云以及使用预定义的度量标准/目标函数将其分开，来识别有意义的分组，也称为类或聚类。此外，聚类过程结束，当完成时，在一个特定的类拥有的像素数据，一般具有高的相似性而类之间的数据的关联性较低。聚类最大的优点是对于他人简单而易于实现。然而，由于点云都是图像依赖的，鉴于图像处理是一个具有挑战性的任务，选择/初始化聚类数目，以获取语义化分。此外，随着特征空间维数的增加呈指数增长，获得明确的聚类数成为一个艰巨的任务。尽管有许多聚类方法开发用于各种应用，使用特定的固定点集合划分一个特征空间是最广泛采用的方法。泰森多边形法曲面（VT）是一种方法，它将特征空间分解为各种类（称为泰森多边形单元/区域），使用固定的称为站点的点集，这样特征空间中的每个观测值通过一定的距离度量方法分配到最近的站点。具体点说，如果 X 是特征空间，距离约束函数为 d，$(P_k)_{k\in K}$ 是特征空间中 K 个站点的集合，然后形成泰森多边形单元 V_k，满足如下约束：

$$V_k = [x \in X \mid d(x, P_k) \le d(x, P_j) \forall j \ne k] \tag{B.1}$$

其中 $d(x,P_k)$ 表示 x 到 P_k 的距离。Arbelaez 等人提出了一个基于 VT 的使用颜色和光照的图像分割技术。分割目标包括两步：1. 预分割， 2. 层次表示。预分割使用 VT 处理，其中光照通道组件作为站点形成极端镶嵌的泰森多边形区域。第二步从极端镶嵌的泰森多边形区域做一个分层划分，使用伪距离度量，称作超度量，在彩色图像中详细定义。随后，一个实软边界图像称为超指标等值线图（UCM）到最后构造分割。质心的泰森多边形法曲面是 VT 的一个特殊分类，在其中产生泰森多边形单元的站点被选择质心。Wang 等人扩展基础的 CVT，集成边界相关的能量函数，使用经典的聚类能量度量，提出边界权值的质心泰森多边形曲面（EWCVT），分割彩色图像很有效。CVTs 形成许多好的聚类算法的核心，比如 K-means。K-means 算法将 n 个像素划分为 K 个类，通过最小化目标函数。从彩色分割的角度来看，上述算法分析了三维空间的图像数据，最终区分 K-个站点（聚类中心），这样，最小化平均平方距离其最近的每个数据点。对于这个效果，在一个任意迭代（称为泰森多边形迭代或劳埃德算法）中，整个颜色空间通过分配每个观察值到最近的聚类中心，被划分为 K 个区域。在这之后，重新计算聚类中心的一个新的估计，作为下一次迭代的聚类中心。算法迭代直到收敛。McQueen 是第一个使用 K-means 算法处理多变量数据的人。最近的研究中，Kanungo 等人突出一个有效的方法称作“滤波算法”，使用 kd 树表示图像数据。对于树上的每一个节点，一组候选中心通过过滤传递到节点的子节点。这个方法的最大优点是，kd 树通过原始数据形成，而不是计算的中心形成，对比于传统的 K-means 架构，它不授权在所有迭代中的结构更新。Chen 等人使用一般化了的 K-means 算法，因自适应聚类（ACA）而著知，具有与颜色和纹理相关的空间适应特征，产生感知上的协调的分割。因此，ACA 算法是规范空间盲目聚类的一个例外。在它的研究中，Mignotte 提出了一个新的彩色图像分割程序，基于多类最小化欧拉距离函数的 K-means 聚类结果的融合，从六个不同的彩色空间——RGB、HSV、YIQ、XYZ、LAB 以及 LUV——描述图像。一旦获取了颜色空间的标签域，每个像素计算局部类标签的直方图，所有直方图都作为特征向量成为融合机制的最终输出。使用 Bhattacharya 相似系数，融合机制与包含 K-means 算法，这是一个基于直方图的相似度量。Mignotte 使用空间约束的 K-means 标定过

程替代融合机制，达到最优的结果。而之前 Mignotte 开发的算法，目的是寻求可能的集成的多个从简单数据模型获得精确结果的分割图，之后的算法在某种意义上是新颖的，在 K-means 框架隐含图像的空间关联，从而找到最好的解决方案，因此整个过程不是空间盲目的。

均值平移聚类是另一个常用的聚类方法，过去十年普遍适用到灰度/彩色图像分割中。它通常使用无参数技术促进使用任意形状聚类的多维特征空间，基于 Fununaga 等人提出的“均值平移”概念。均值平移过程是一种核密度估计（或 Parzen 窗口技术），审查特征空间作为一个经验概率密度函数（pdf），并考虑从任意图片中离散采样的像素值的集合。这个过程利用了这样一个事实，特征空间聚类/密集区域通常表现为上述的 pdf 模型。下面，如果 S 是 n 维欧拉空间的有限点云，X 与 K 是一个具有特定属性的对称核函数，在像素 $x \in X$ 中使用相邻的点的权值组合计算的样本均值 m(x)：

$$m(x) = \frac{\sum_{s \in S} K(s-x)s}{\sum_{s \in S} K(s-x)} \tag{B.2}$$

为此目的，在每个位置为 x 的像素，均值平移向量 $m(x)-x$ 通过在 x 处的中心获得，因此向量点朝向最大的强度增加方向。接下来，$x \leftarrow m(x)$ 表示将 x 往均值 m(x) 方向平移。重复这个过程知道 m(x)达到收敛。均值平移最后，每个像素达到最近的 pdf 峰值。因为均值平移算法使用空间信息，它是传统的空间盲目聚类的一个例外的代表。均值平移聚类通过边缘信息作为向导，这可以从 Christoudias 等人的工作中看到，他们提出边缘检测与图像分割（EDISON）系统，目的在于提高在保持或理想最小化图像过分割提取相同区域的灵敏度。如图 B.5,显示了一些使用默认参数的 EDISON 系统的结果（空间带宽 h_s ，颜色带宽 h_r ，最小区域尺寸 M=20）。Hong 等人提出了均值平移分割算法的改进版本，通过合并的方法：1. 加强的模式识别技术，2. 全局分析局部识别模型的优化过程，以及 3. 纹理区域估计以在各种背景条件下获得稳定的结果。Ozden 等人首先使用了一个有效的技术，将低级别的颜色与空间及纹理特征组合到均值平移框架中用于图像分割。

基于神经网络的数据聚类是一个唯一起源于人工智能的类别。在这个领域，自组织映射方法（SOMs）在过去十年被广泛关注。一个自组织图或自组织特征映射（SOFM）——两者之一被著知为 Kohhoen——是一个特定的人工神经网络类别，首先被 Kohonen 引入作为提供智能的高维/多维特征空间到低维空间的映射。SOM（如图 B.6）由以维度与输入特征空间相同的向量形式组织的神经元的输入层组成。每个神经元并行的连接到二维的矩形或六边形分布的神经元的输出层，而且对应相邻的神经元使用适当的权重描述不同长度的连接。SOM 操作在训练时期，使用特征空间的采样子集逐渐地建立特征空间映射，在映射之后，一个任意的新输入将自动被分类。在两个模式操作的顶点，一个低维的映射是后续产生的，低维映射反映特征空间样本的拓扑关系。换句话说，输入特征空间中具有相同特征的样本在映射中形成不同的类。

Huang 等人救彩色图像分割提出了方法论，使用一个两个阶段的基于 SOM 的 ANN。算法初始化是将输入图像从 RGB 到 HVC 颜色空间的转换，被用于 SOM 去鉴别一大堆的颜色类初始集。结果的类的集合进一步通过计算归一化的欧拉距离提炼，得到的类间距离用于第二次 SOM 的输入，SOM 鉴别最后的一批分割的类。相同的方法，Ong 等人基于层次的两个阶段的 SOM 构建了一个彩色图像分割技术，第一阶段识别输入以 Luv 空间描述的图片的主要颜色，第二阶段集成可变大小的一维特征映射以及类合并/丢弃操作以获得最终分割结果。Li 等人使用 SOM 和模糊 C 均值聚类（FCM）过程展示了一个有效的彩色图像分割方法。算法初始时使用 SOM 在五种颜色空间中查找非常合适的图像独立的特征。接着，独立的特征应用到 FCM 算法获取输出的分割结果。Dong 等人建立了两个交替的基于 ANN 的图像分割策略。第一个策略是无监督的，包括使用基于 SOM 学习 Luv 彩色空间的输入图片区分出一系列的颜色原型。这些颜色原型传递到仿真的热处理驱动的聚类程序中产生非常好的聚类。第二种方法，构建了前面提到的算法，组合了层次的像素学习（产生了场景中不同大小的颜色原型）和分类协议，以提供更高精度的分割效果，这是监督学习的风格。在 Teo 等人的工作中，首次看到使用 SOM 和自适应共振理论（ART）划分彩色图像，他提出了两种混合 ANN 架构称作 SOMART 和 SmART，

与传统的基于 SOM 的技术相比，提高了分割效果。另一方面，Araujo 等人设计了一个快速的鲁棒性好的自适应拓扑 ANN 模型，称为局部适应容纳域 SOM（LARFSOM），基于在网络训练阶段使用小比例的输入图片像素快速学习的颜色分布，能推理紧密的聚类以及适当的聚类数。算法用于各种不同分割复杂度的彩色图像，证明胜过一些先前的基于 SOM 的技术。Frisch 引入了一种新方法，对光照变化鲁棒性强，使用了 SOMs 来构建模糊测量，可适用于彩色图像分割。这项工作是在有效的模糊测量中一个独一无二的尝试，用于完成了图像分割任务，最初起源于使用基于 SOM 的处理。Ilea 等人设计了一个完全自动的图像分割算法叫 Ctex，使用颜色和纹理描述符。Ctex 分割架构使用 SOM 分类器首先从 RGB 和 YIQ 颜色空间的输入图像中提取主要的颜色。在这过程中，合适的聚了数（K）也知道了。接着，在 6 维的跨越包括上述陈述的颜色空间的多空间使用传统 K-means 聚类算法，纯粹通过颜色信息来获取分割结果。接下来，使用 Gabor 滤波器带计算纹理特征，与先前取得的分割一样，都作为新的适应的空间的 K-means（ASKM）聚类算法的输入，聚类算法描述输入图像中颜色和纹理的连贯区域。

聚类方法讨论到目前为止都是典型分为硬聚类方法，因为特征空间的每个观察值都有个独特的强制的聚类分配，产生的类都有尖锐的边界。作为对比，模糊聚类做了重要的工作，它促进观察值承受一个可能的隶属度或隶属成员，而不是只属于一类，导致重叠的类以及具有“软”边界的类。模糊 C 均值算法最先被 Dunn 开发，是最广泛使用的模糊聚类方法，其技术与 K-means 类似，通过最小化目标函数，划分一个集合成 n 个像素集（ $X=\{x_1,x_2,...x_n\}$ ）到模糊类 C（ $C=\{c_1,c_2,...,c_n\}$ ）。FCM 算法使用的目标函数为

$$J_m=\sum_{i=1}^{n}\sum_{j=1}^{c}u_{ij}^{m}\left\|x_i-c_j\right\|^2 \tag{B.3}$$

其中

$$u_{ij}^{m}=\frac{1}{\sum_{k=1}^{c}\left(\frac{\left\|x_i-c_j\right\|}{\left\|x_i-c_k\right\|}\right)^{2/(m-1)}} \quad \text{以及} \quad c_j=\frac{\sum_{i=1}^{n}u_{ij}^{m}x_i}{\sum_{i=1}^{n}u_{ij}^{m}} \tag{B.4}$$

从公式（3）和（4）中可以推断 FCM 目标函数与 K-means 的不同，对于特征空间中不同的观测值 x_i ，FCM 合并成员值 u_{ij} ，以及“模糊”项 $m \mid m \in \{1 \le m \le \infty\}$ 导致了类的模糊性扩展。

在它们的工作中，Yang 等人提出了两个基于特征值模糊聚类方法，分别是分离特征空间 FCM（SEFCM）和对偶特征值 FCM（CEFCM），对于彩色图像的目标分割具有期望的属性。考虑一幅任意输入图像，输入图像有预定义的像素集，其中的颜色空间初始化时使用组成成分分析法划分为两个称作主要的和残留的特征空间。在这之后，SEFCM 设计获得一个分割输出，通过集成独立使用 FCM 算法的结果。集成的过程包括从两类聚类结果中通常像素的逻辑选择。另一方面，CEFCM 分配通过考虑主要的和残留的特征空间获得输出分割结果。这些算法都被显示优越与传统的 FCM 聚类算法，就彩色图像分割方面而言。Liew 等人通过使用上下文信息加入空间一致性研究了一个自适应的模糊聚类算法。这个方法目的在于利用大部分传统意向中内部像素之间的关联性到模糊框架中。Chen 等人提出了高性能的 FCM 算法版本，使用了两阶段，包括数据下采样和聚类。这个更有效的计算方法产生的结果与传统的 FCM 方法类似。更近的，Hung 等人开发了一个带权值的 FCM（WFCM）聚类技术，在其中不同特征的权值使用引导方法。合并引导方法为每个统计不同观测点的特征提供了满意的权值，提高了 WFCM 程序的性能。Tziakos 等人提出了一个使用拉普拉斯特征（LE）映射算法，流形学习技术，来推动 FCM 算法的性能。这个方法首先通过重叠区域在一个高维空间提取局部图像特征，按这种方式，通过图谱理论学习一个低维的副本。接下来，使用基于 LE 的维度下采样技术计算抓获了局部图像特征变化的低维映射，最终常常提高 FCM 性能。Krinidis 等人以及 Wang 等人开发了一个其它的有效的 FCM 版本，同时使用了局部强度与空间信息。Yu 等人建立了蚁群移植的模糊 C 均值聚类混合算法（AFHA）用于彩色图像的分割，使用了智能聚类中心初始化自适应聚类的图像元素分割以及使用了子采样提高计算效率。AFHA 结构的记过具有更小的失真以及更稳定的聚类中心，相比于传统的 FCM 算法。

除了这节已经讨论过的，图像分割中一些特殊的聚类方法也被提出。Veenman

等人为聚类开发了一个有效的优化的模型，使用多细胞联合进化算法（CCA），这不需要任何的先验知识，不需要知道聚类数目。在另一方面，Allili 等人研究了将聚类模型与高斯混合模型结合起来，以及相关的特征选择，缓和欠分割与过分割问题。Jeon 等人一种彩色图像的稀疏聚类的方法，使用正张量分解（PTF）。Aghbari 等人提出了希丘陵操纵的处理，分割任意的彩色图像就等同于在对应的 3-D 强度直方图上寻找丘陵。Ma 等人引入使用有损失的数据编码和压缩后聚类，他们在自然风景彩色图片上测试了他们的工作。Ma 等人使用的算法是由 Yang 等人提出的，Yang 等人提出了基于压缩的纹理合并（CTM），将分割任务认为是聚类纹理特征，这些特征按照混合高斯模型模型分布，在其中的聚类中，数据是压缩的数据。CTM 算法的样本分割结果使用默认参数（编码数据长度 γ =0.2），这些显示在图 B.7 中。Huang 等人提出了纯粹对彩色图像分割有效的“先聚类然后标定”的方法。

直方图阈值 直方图阈值是空间盲目的技术，最初基于通过甄别强度直方图中峰值、山谷以及形状来分割图像。与聚类类似，直方图阈值与其他分割算法最大的区别就是不需要将要分割图像的任何先验信息。由于它的清晰性，强度直方图阈值初始时在分割灰度图中非常流行。然而，在分割过程中发现，对于低对比度的图像阈值强度直方图工作得并不那么好，这些图没有明显的峰值，在有峰值噪声的时候，产生模糊不清的区域。除此之外，对于彩色图像，多维空间的阈值确定是一个很困难的工作。图 8 显示了海星和船图片的 RGB 颜色直方图，这些结果使用开源的 ImageJ 插件称作颜色监视 3D，来获得。3-D 直方图中的每个颜色维度使用一个球体表示，球体的容量与颜色出现的频率成正比。从这些直方图中，可以观察发现，寻找多个阈值来有效的分割是一个极具挑战的工作。

Kurugollu 等人提出了一个彩色图像分割算法，包括两个主要步骤，称作多阈值与融合。这种方法初始使用成对带组合（RG，GB 以及 BR）形成 2-D 直方图，每个都从属于峰值查找的方法。接着，基于描绘的峰值，用一个多阈值的计划形成与每个通道不同的分割结果。以上的三个分割结果使用标签一致算法融合，然后空间彩色主体滤波器产生最终的分割结果。在一个类似的框架中，Cheng 等人设计了一个基于直方图阈值的彩色图像分割方法，其沿着相邻的具有相同值的像素，同步捕

获出现的灰度值。分割例程初始时通过一个每个颜色通道独立形成颜色直方图，直方图的峰值用于后续的阈值计划来获取初始的过分割类集。最后，从红色、绿色和蓝色板上获得三个聚类结果，分别的，结合形成最后的分割结果。Mushrif 等人采用直方阈值的概念，基于粗糙集合理论来设计一个有效的彩色图像分割算法。直方被定义为可能为特定分割部分的像素的集合。他们的三步架构包括计算直方，接着使用阈值，最后是区域合并过程（在 2.2.2.1 节中讨论）。除此之外，他们提高了上述方法，尽管这个方法的工作是 Mushrif 等人做的。Mushfrif 等人使用一个 Atanassov 的直观模糊集（A-IFS）直方表示输入图片。在他们的工作中，Manay 等人提出了自适应阈值结构，使用各向异性扩散模型进行快速分割，在任意的区域，使用最优的阈值能接近图像边缘。

（图略）

图 B.1 伯克利分割基准:(a) 原始图像, (b) 到 (f) 多个自动产生分割的区域边界

（图略）

图 B.2 图像分割算法的高级别分类

（图略）

图 B.3 图像分割算法的低级别分类

（图略）

图 B.4 简单的图像及其 3-D 点云

（图略）

图 B.5 EDISON 系统的结果

（图略）

图 B.6 矩形神经网络层的自组织图（SOM）

（图略）

图 B.7 CTM 算法的结果

在 学 取 得 成 果

一、 在学期间所获的奖励

无

二、 在学期间发表的论文

无

三、 在学期间取得的科技成果

无

致　　谢

本科毕业论文的设计的思想启蒙及相关材料大部分是北京科技大学计算机与通信工程学院物联网与电子工程系的刘磊老师提供的，刘老师作为我毕设的指导老师，一直孜孜不倦地指导着我的毕业设计，这里要非常感谢刘老师的帮助。

我要感谢我身边一起做毕设的同学，他们是吴翔、阮光中、吴曙东和尹晓龙，感谢他们无私的帮助，在每周的毕设任务进度报告会上，他们都教会了我许多与本文工作密切相关的知识。感谢他们参与讨论，这给了我一个可以在互相交流中汲取知识的环境。

我还要感谢开源社区以及开源社区里的朋友们，开源社区的开源工具或代码免去了我大量的资金与时间的投入，开源社区的软件开发思想尤其是 Linux 的哲学思想影响着本文设计框架的搭建过程，受益颇多。