基于数学形态学的聚类集成算法

罗会兰[1,2] 危 辉[1]

(复旦大学智能信息处理重点实验室 上海 200433)[1] (江西理工大学信息工程学院 赣州 341000)[2]

摘 要 提出了基于数学形态学的聚类集成算法CEOMM。它利用不同的结构元素的探针作用,对不同的结构元素探测出来的簇核心图进行集成,在集成所得到的簇核心基础上聚类。实验结果表明,算法CEOMM对有复杂类形状的数据集进行聚类时,效果比传统聚类算法更好,且能确定聚类数。而且由于采用了不同的结构元素进行探测,对于由不同形状的类构成的数据集其聚类效果很理想。

关键词 聚类集成,数学形态学,结构元素

中图法分类号 TP181 **文献标识码** A

Clustering Ensemble Algorithm Based on Mathematical Morphology

LUO Hui-lan[1,2] WEI Hui[1]

(Key Laboratory of Intelligent Information Processing, Fudan University, Shanghai 200433, China)[1]

(School of Information Engineering, Jiangxi University of Science and Technology, Ganzhou 341000, China)[2]

Abstract In this paper, a clustering ensemble algorithm named CEOMM was proposed, which combines multiple clustering cores explored by different structure elements to get a desirable and correct clustering core of a data set. And then CEOMM gets the clustering of the data set based on the ensemble clustering core. Experimental results demonstrate CEOMM can cluster data with complex cluster shapes better than the classical clustering algorithms, and it can also find an optimal number of clusters. Moreover, CEOMM can discover overlapping clusters with different arbitrary shapes, because it uses different structure elements.

Keywords Clustering ensemble, Mathematical morphology, Structure elements

1 国内外研究现状

数学形态学基于代数学、集合论和几何定理,是图像分析的一个分支[1],最初是由Matheron(1975)[2]和Serra(1982)[3]为实现任意集合的大小改变而发展起来的。数学形态学主要应用于图像识别领域,将数学形态学作为工具,从图像中提取对于表达和描绘区域形状有用的图像分量,比如边界、骨架以及凸壳,等等[4]。在文献[5]中,McReynolds将数学形态学操作分水岭用于图像分割。在文献[6]中,BREEN给出了形态学工具的一个综述,并用大范围的实际图像分割例子来证明它们的有效性。在数学形态学应用中,一个图像定义为像素的集合,每个像素将图像所在的坐标映射到一个值,比如灰度图像中的灰度值、二值图像(Binary images)中的0或1,或彩色图像中的三色值矢量。

Postaire等(1993)在文献[7]中将数学形态学用于聚类分析,将d维实数空间中的对象映射到一个d维整数空间(离散空间)中,从而将实数数据集转换成图像格式,然后利用形态运算开(open)和闭(close)进行聚类分析。但是文献[7]中并没有就如何选择离散网格空间的大小,如何根据先验知识选择结构元素做出论述,对于形态运算的应用也仅限于一次开运算后跟着一次闭运算,对于运算后所得到的集合如何进行聚类分析没有做出论述。V Starovoitov(1996)在文献[8]中提出了一种变形的膨胀(dilation)和腐蚀(erosion)运算,并且提出了一种确定离散网格空间大小的方法,但同样对运算后所得到的集合如何进行聚类分析也没有做出论述。在文献[9]中邸凯昌等提出用圆形结构元素逐渐增大尺寸来进行闭运算从而自动确定类数的启发式方法。在此文献中他们绘出了圆形结构元素的大小对相应大小结构元素闭运算后产生的簇个数的图,发现在相应正确类数之处有一个突然的下降。在文献[10]中Ferreira Costa等结合SOM(self-organizing map)与数学形态运算对形状复杂的簇进行聚类。SOM(self-organizing map)用于高维数据的可视化或将数据信息进行压缩,将它用于聚类时,可以将输出单元设置成聚类数,也可以利用一些数据的先验信息来设定一个二维输出层。Ferreira Costa等利用数学形态学操作分水岭(watershed)对SOM输出的二维单元进行聚类。他们首先用完整待聚类数据集对SOM进行训练产生二维单元,然后在其上用分水岭(watershed)操作对这些单元进行聚类标记。再对聚类产生的每个簇中的数据子集训练SOM,同样用分水岭(watershed)对这些子SOM二维单元进行聚类标记。重复此过程从而得到数

到稿日期:2009-09-04 返修日期:2009-12-07 本文受国家973项目(No. 2010CB327900),国家自然科学基金(No. 60303007),上海科技发展基金(No. 08511501703),上海市智能信息处理重点实验室开放课题(No. IIPL-09-009)资助。

罗会兰(1974—),女,博士后,副教授,主要研究方向为机器学习、模式识别,E-mail:luohuilan@sina.com;危 辉(1971—),男,博士,教授,主要研究方向为认知模型、计算机视觉。

据的一个分层结构图。

在聚类分析中，研究对象是用多维特征值向量表示的。聚类分析就是将对象集(数据集)根据其相似性进行划分，使得同一类或簇内的对象比较相似，而不同类间的对象不相似。一些传统的聚类方法如 k-means 和 k-mediods 只能识别球形且大小相似的簇，在识别任意形状的簇时效果不理想。为了识别形状复杂的聚类，有许多算法被提出，比较经典的有 CURE[11]，DBSCAN[12]和 Chameleon[13]，还有一些基于 SOM 的二层算法也能识别一些形状复杂的聚类[10,14]，但是这些算法都比较复杂和难以理解，很难将数据的先验信息和领域知识融合其中。由于聚类分析是一种探测型数据分析工具，这也意味着可以利用数据的各种不确定信息来分析和验证。而数学形态学操作就是一种很好的探测型工具，可以用不同形状和大小的结构元素对数据进行不同的形态学操作，以获得数据的不同方面的核心信息，从而验证关于数据的猜想。但是数学形态学操作是为图像分析定义的，要将它应用于表示成多维矢量的对象数据，还需要将数据转换成适合于形态学运算的格式。

首先在第 2 节简单介绍了常用的数学形态学操作。然后在第 3 节介绍了文献[15]中提出的一个基于数学形态学分级操作系列的聚类算法。在此基础上，在第 4 节将数学形态学与集成技术结合起来提出了一个基于数学形态学的聚类集成算法，此算法首先利用不同的结构元素产生不同的数据核心信息，然后对这些不同核心信息进行集成，基于此集成得到的核心对原始数据集进行聚类。实验表明它的效果非常好，对于包含形状复杂且各异的簇的数据集也能正确聚类和得到正确的类个数。最后总结了本文。

2 基本的数学形态学运算

这里提及的数学形态学运算针对的是二值图像形态学。所谓二值图像是指那些灰度只取两个可能值的图像。这两个灰度值通常取为 0 和 1。习惯上认为取值为 1 的点对应于景物中的物体，表示成黑色。而取值为 0 的点构成背景，在图像中显示成白色。这里介绍了本文中用到的一些最基本的数学形态学运算。其它的一些复杂的数学形态学操作或算法都能用它们定义出来，这里就不做介绍。

最简单的运算是膨胀和腐蚀，基于这两种运算定义了开和闭运算。下面的定义参考了文献[16-20]。

令 X 和 T 为 Z^d 的子集，其中 Z 表示整数集。

定义 1 X 平移 s，或称 X 被 s 平移，用 X_s 表示，定义如下：

$$X_s=\{t\in Z^d; t=x+s, x\in X\} \tag{1}$$

定义 2 X 被 T 膨胀运算，用 $X\oplus T$ 表示，定义如下：

$$X\oplus T=\{e\in Z^d; e=x+s, x\in X, s\in T\} \tag{2}$$

$X\oplus T$ 相当于 X 平移 T 中所有元素后取得的并集，膨胀使集合 X 变大，其中 T 称为膨胀的结构元素。

定义 3 X 被 T 腐蚀运算，用 $X\Theta T$ 表示，定义如下：

$$X\Theta T=\{f\in Z^d; x=f+s, x\in X, \forall s\in T\} \tag{3}$$

这个定义表明，若 f 是 $X\Theta T$ 中的点，则它一定满足这样的性质：结构元素 T 被 f 平移后得到的集合应当包含在 X 内。腐蚀使集合 X 变小，其中 T 称为腐蚀的结构元素。

定义 4 使用结构元素 T 对集合 X 进行开操作，用 $X\circ T$ 表示，定义如下：

$$X\circ T=(X\Theta T)\oplus T \tag{4}$$

开运算的定义说明，若 p 是 $X\circ T$ 中的点，则一定存在某个 y，使 T 平移 y 后所得集合包含在 X 中，而 p 又恰在 T 的新位置中。因此，$X\circ T$ 表示 X 中恰好包括结构元素 T 的那些部分的残迹。

因此，使用 T 对 X 进行开操作就是先用 T 对 X 腐蚀，然后再用 T 对结果进行膨胀。开操作一般使对象的轮廓变得光滑，断开狭窄的间断和消除细的突出物，消除散点和“毛刺”。

定义 5 使用结构元素 T 对集合 X 进行闭操作，用 $X\cdot T$ 表示，定义如下：

$$X\cdot T=(X\oplus T)\Theta T \tag{5}$$

使用 T 对 X 进行闭操作就是先用 T 对 X 膨胀，然后再用 T 对结果进行腐蚀。闭操作同样使对象的轮廓变得更为光滑，但与开操作相反的是，它通常填充或者说消弥狭窄的间断和长细的鸿沟，消除小的孔洞，并填补轮廓线中的断裂。在选择了适当的结构元素后，可以通过闭运算将两个邻近的目标连接起来。

3 基于数学形态学分级操作系列的聚类算法 COHMMOP

我们在文献[15]中提出了一种简单的基于数学形态学分级操作系列的聚类分析算法 COHMMOP(Clustering based On Hierarchical Mathematical Morphology Operation Procedures)：首先在结构元素和离散网格空间的大小确定方面充分考虑到关于聚类对象集的先验知识；其次使用多次具有不同结构元素的膨胀和腐蚀运算操作系列；再在此基础上利用膨胀和交(intersection)运算来实现连通分量的提取，从而实现在离散空间的聚类。

3.1 离散化参数的选择及离散化

(1)离散化参数的选择

假设待聚类数据集 $X=\{x_1, x_2, \cdots, x_n\}$ 中的任一数据 x_i 用 d 维向量表示成 $x_i=[x_{i1}, x_{i2}, \cdots, x_{ij}, \cdots, x_{id}]^t$。将数据集 X 中每一数据的每一个分量如 x_{ij} 映射到一个特定整数区间 $[1, m_j]$ 内的一个整数，就称为数据集的离散化。而这也就意味着有多少维，就需要确定多少个参数。在文献[7]中将每一维都映射到相同的整数区间 $[1, m]$ 内，而在文献[8]中提到了一种确定各个维的参数 m_j 的算法，则把这个问题转换成求在实数空间中超立方体的尺寸大小，通过改变超立方体的尺寸来计算一个代价值，从而确定各个 m_j。

本文采用各个维相同的参数的方法来逐步改变 m 的值，当所得到的聚类数不变时，就是所要的值。根据文献[7]，稳定的 m 值一般位于 20 到 60 之间。

(2) 离散化

文献[7]虽然谈到了离散化，但并没有详细论述，这里的离散化过程参考了文献[19]。离散化过程分为两步。第一步首先对数据集 X 中的所有向量进行规范化，使向量的所有分量位于 $[1, m]$ 范围内。设 X 中有 n 个向量，m 是离散化参数，这种规范化可以通过下面的变换得到：

$$y_{ij}=\frac{x_{ij}-\min_{q=1,\cdots,N}x_{qj}}{\max_{q=1,\cdots,N}x_{qj}-\min_{q=1,\cdots,N}x_{qj}}\times m, i=1,\cdots,n;\ j=1,\cdots,d \tag{6}$$

式中，x_{ij} 表示 i 向量的第 j 个分量。用 S 表示这一步规范化所得到的结果集：

$$S=\{y_i\in[1,m]^d, i=1,\cdots,n\} \tag{7}$$

接下来，把 $[1,m]^d$ 离散化为 m^d 个超立方体，然后识别向量 $y_i\in S$ 所在的超立方体。这可以通过取 y_i 的每个坐标的整数部分来得到，所得到的结果向量可作为超立方体的识别标签，用 z_{ij} 表示如下：

$$z_{ij}=[y_{ij}], i=1,\cdots,n, j=1,\cdots,d \tag{8}$$

式中，$[y_{ij}]$ 表示 y_{ij} 的整数部分。令 $S1$ 是包含所有新变量 z_i 的集合，删除所有重复向量后得到该集合。因此，$S1$ 中的每一个元素都对应于一个非空的超立方体，这个非空超立方体所在位置的像素的灰度值为 1，而其它空单元的灰度值为 0。至此就完成了离散化过程。

3.2 膨胀和腐蚀运算操作系列的选择及运算

在离散化后所得到的数据集 $S1$ 上进行数学形态运算，以将各个类分离开来实现有效聚类。

3.3 离散空间 $S2$ 的聚类

使用一系列膨胀和腐蚀形态运算后，得到了一个新的离散数据集 $S2$，这个数据集有效地去除了孤立点和空洞，把各个类很好地隔离开来了，在这个离散数据集 $S2$ 上进行聚类分析，只需确定连通分量，每个连通分量对应一个类的核心，找到所有的连通分量后就完成了在这个数据集 $S2$ 上的聚类分析，一个连通分量就是一个类，连通分量的提取采用下面的算法[20]：

Step1　在 $S2$ 中任取一个还没有被聚类的点 p，令 $C_0=\{p\}$；

Step2　while $C_k\neq C_{k-1}$

$k=k+1; C_k=(C_{k-1}\oplus T)\cap S2$，其中 T 是一个适当的结构元素。

end

包含 p 的连通分量就是 C_k，并将 C_k 中的所有点标记为同一类。

Step3　如果 $S2$ 中还存在没有标记的点，转到 Step1；否则算法结束。

其中结构元素 T 的选择决定所提取的连通分量是八连通还是四连通的。如果选择的结构元素有八连通性，则产生的连通分量是八连通意义下的，如果选择的结构元素是四连通的，则产生的连通分量是四连通意义下的。在实验中应用了 matlab 7.0.1 自带的一个标记连通分量的函数来标记连通分量。

3.4 基于在 S2 中的聚类求出实数空间向量集 X 中的聚类

将原实数空间的向量集 X 中的向量聚类分两步[19]：第 1 步，对于所有向量 $x_i\in X$，如果相应的 $[y_i]\in S2$，则将 x_i 向量分配给 $[y_i]$ 所属的类。令 X_1 是这一步中分配了类的向量的集合，也称为核心点集合。第 2 步，对于 $X-X1$ 中的每个向量，将其分配给离它最近的核心点所在的类。

3.5 COHMMOP 算法小结

COHMMOP 算法是一种基于数学形态学的聚类分析方法，它将用于图像处理的方法引入聚类分析，得到了理想的效果[15]。它首先将数据规范化，从而消除了样本特征值（分量）之间数据度量大小、类型等不协调的负面影响；然后通过一系列的形态运算将各个类很好地隔离开来，从而达到好的聚类效果。这种方法虽然能很好地利用各种先验知识，并且能自动确定聚类数，但是在没有这种知识存在的情况下，可能需要多次实验，才能得到好的离散化参数、结构元素和形态学操作运算。

4 基于数学形态学的聚类集成算法 CEOMM

由于 COHMMOP 算法[15]需要用户精心地选择离散化参数、结构元素以及运算系列，在对数据集没有先验知识以及没有专家指导的情况下，如何能减少聚类风险，提高平均聚类性能，对此我们提出了基于数学形态学的聚类集成算法 CEOMM (Clustering Ensemble based On Mathematical Morphology)，即用不同的离散化参数、结构元素或数学形态学运算系列得到一个簇核心集体，对此簇核心集体进行集成，用基于集成所得的簇核心来确定类的个数并完成最终聚类。

4.1 簇核心（或簇代表）集体的生成

传统的聚类集成方法都是先生成一个聚类集体，然后在其上集成得到数据集的一个聚类。生成聚类集体的方法有很多，有利用不同数据子集、不同特征子集、不同算法、不同算法的初始参数等等来生成一个聚类集体。一般来说聚类集体中的成员聚类都是原始待聚类数据集的一个完整聚类。CEOMM 与传统的聚类集成方法的不同之处在于，CEOMM 并不是生成一个聚类集体，而是生成一个簇核心集体。也就是说 CEOMM 首先使用不同的离散化参数，或不同结构元素，或不同数学形态学操作系列生成不同簇核心，然后构成一个簇核心集体。

下面举例说明生成簇核心集体的方法。其中每个簇核心成员采用不同的结构元素来生成。

图 1 是一个有 5 个形状各异的类的数据集。先对图 1 的数据源使用离散化参数 $m=60$ 将其离散化，离散化后的数据如图 2 所示。然后使用 12 种不同的结构元素对其进行开运算加闭运算。12 种不同的结构元素分别是：

(1)长度为 6，倾斜角度为 30 度的线形结构元素 strel('line',6,30)；

(2)半径为 2 的圆盘形结构元素 strel('disk',2)；

(3)半径为 3 的圆盘形结构元素 strel('disk',3)；

(4)半径为 4 的圆盘形结构元素 strel('disk',4)；

(5)半径为 1 的钻石形结构元素 strel('diamond',1)；

(6)半径为 2 的钻石形结构元素 strel('diamond',2)；

(7)边长为 2 的正方形结构元素 strel('square',2)；

(8)边长为 3 的正方形结构元素 strel('square',3)；

(9)边长为 4 的正方形结构元素 strel('square',4)；

(10)边长分别为 2 和 3 的长方形结构元素 strel('rectangle',[2,3])；

(11)边长为 3 的八角形结构元素 strel('octagon',3)；

(12)任意形状结构元素[1 0 0;1 0 0;1 0 1]。

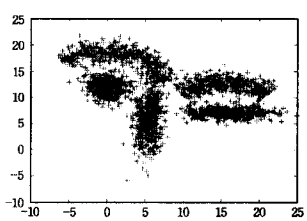


图 1　有 5 个形状各异的类的数据集

分别使用上述这 12 种不同的结构元素对离散化后的数据进行开运算加闭运算后得到的不同簇核心，如图 3 所示。可以看出，有的结构元素得到的效果不理想，也没有找到正确的聚类数，而有的结构元素得到的效果很好。

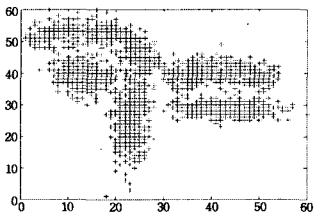
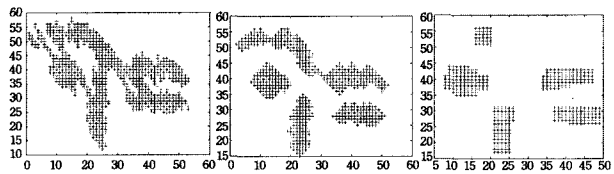
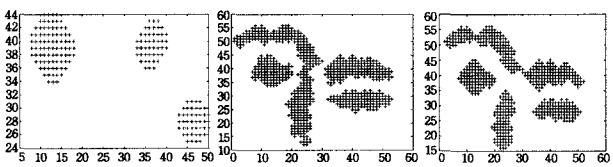


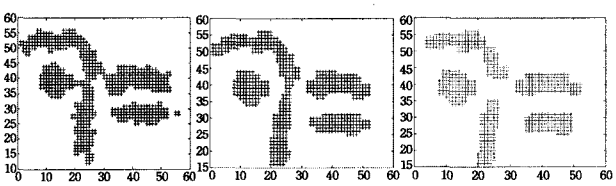
图 2　图 1 所示的数据源使用离散化参数 $m=60$ 离散化后的效果图



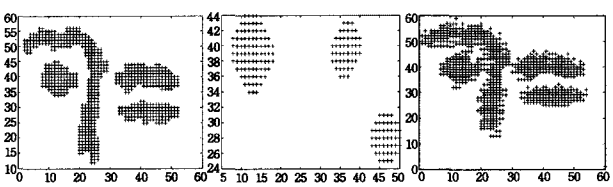
(1) strel('line',6,30)；　(2) strel('disk',2)；　(3) strel('disk',3)



(4) strel('disk',4)；　(5) strel('diamond',1)；　(6) strel('diamond',2)；



(7) strel('square',2)；　(8) strel('square',3)；　(9) strel('square',4)；



(10) strel('mectangle',[2,3])；　(11) strel('octagon',3)；　(12) [1 0 0;1 0 0;1 0 1]；

图 3　不同的结构元素对图 2 所示的离散化后的数据进行开运算加闭运算后得到簇核心效果图

这样就得到了一个有 12 个成员的簇核心集体。下一步就可以利用这个集体来集成找到理想的簇核心。

4.2　簇核心集体的集成

如果把簇核心集体中的簇核心成员当成一般的聚类成员，也就是说每个簇核心成员中的每个连通分量作为一个簇，则每个簇核心成员可以得到一个聚类，这样就可以用文献中一般的集成方法来集成了。但是我们的算法 CEOMM 采用的方法是直接对簇核心集体进行集成得到一个集成簇核心，也就是希望在簇核心集体的基础上集成产生一个更优的簇核心，基于此集成得到簇核心再进行原始待聚类数据集的聚类。

算法 CEOMM 采用的集成方法很简单，它采用绝大多数投票法对簇核心集体进行集成。比如，在图 7 所示的 12 种簇核心图中有超过一半以上的，也就说大于 6 的成员赞成一个点在簇核心中则将此点保留在簇核心中，否则认为它不属于簇核心。基于此方法对图 3 所示的 12 种簇核心构成的簇核心集体集成后得到的集成簇核心如图 4 所示。从图 4 可以看出，集成后的效果比大多数成员要好，找出各个类的核心了，并且通过找连通分量，也得到了正确的类数为 5。

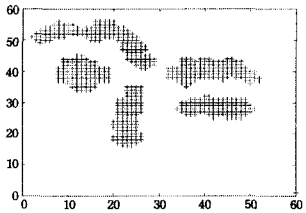


图 4　将图 3 所示的 12 种簇核心用绝对多数投票法得到一个集成簇核心图（或称为簇代表点图）

4.3　利用集成所得的簇核心得到最终聚类

经过集成所有的簇核心成员，得到了一个理想的簇核心，我们将它表示成 $S2$，它有效地找到了各个类的代表点，体现了各个类的形状和大小，并把各个类很好地隔离开来。我们在这个集成所得的簇核心数据集上进行聚类分析，同样是通过确定连通分量的方法来完成，每个连通分量对应一个簇，找到了所有的连通分量就完成了在这个数据集上的聚类分析。在我们的实验中同样采用 matlab 7.0.1 自带的一个标记二值图像连通分量的函数来完成集成簇核心数据集上的聚类分析。

有了簇核心的聚类结果，就可以对原实数空间的待聚类数据集 X 聚类了。首先对于所有 $x_i \in X$，如果经离散化后的相应点 $[y_i] \in S2$，则将 x_i 向量分配给 $[y_i]$ 所属的类。令 $X1$ 是这一步中分配了类的向量的集合，则对于 $X-X1$ 中的每个向量，将其分配给离它最近的核心点所在的类。

图 5 是算法 CEOMM 在图 4 所示的集成簇核心的基础上对图 1 所示的数据集的聚类结果。图 5 中用不同的符号即方形、钻石形、星形、圆形以及加号形表示不同的类。从图 5 可以看出，CEOMM 算法的效果是相当理想的，它能识别不同形状的类。图 6 是 k-means 的聚类效果图，从图中可以看出 k-means 在这种具有形状各异的类且类间隔很小的数据集上的失效。

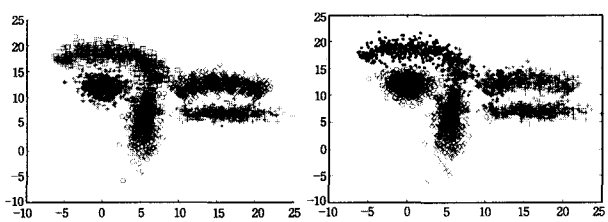


图 5　利用图 4 的簇核心进行聚类得到图 1 所示数据源的聚类结果图（不同形状的符号代表不同的簇）

图 6　k-means 聚类效果图，不同的符号代表不同的簇

4.4　实验

为了进一步说明算法 CEOMM 的有效性，这里给出另一个数据集上的实验结果。如图 7 中(1)子图是一个有 3 个类的数据集，在对其用离散化参数 $m=50$ 离散化后，使用了三个半径为 1 的圆盘形结构元素，两个半径为 1 的钻石形结构元素，两个长度为 2 的正方形结构元素，一个长为 2、宽为 3

的长方形结构元素，两个线形结构元素 strel('line',6,30)和 strel('line',5,0)，以及两个任意形状结构元素 [1 0 0;1 0 0;1 0 1]，总共 12 个结构元素进行开运算加闭运算。对得到的簇核心集体集成后所得的簇核心如图 7 中(2)子图所示。算法 CEOMM 在集成所得的簇核心基础上进行聚类，效果如图 7 中(3)子图所示，得到了理想的效果。而 k-means 的聚类效果如图 7 中(4)子图所示，聚类结果不是很理想。

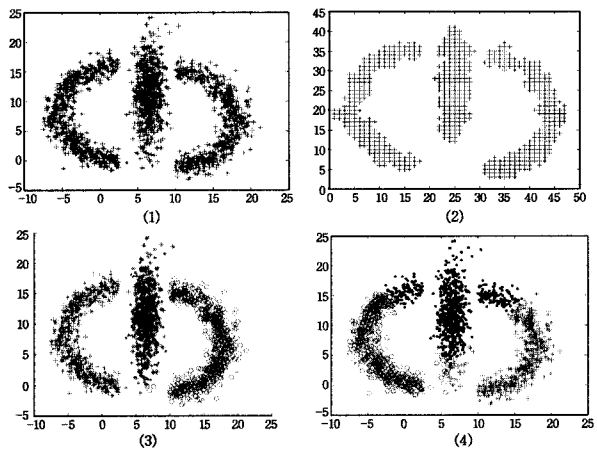


图 7　实验结果：(1) 原始数据集；(2) 集成所得的簇核心；(3) 算法 CEOMM 得到的聚类结果；(4) k-means 算法的聚类结果。

结束语　COHMMOP 算法利用一种形态学操作系列得到被分隔开来的簇核心，并在此基础上确定最终聚类。但是这需要一些先验知识来确定结构元素和离散化参数。所以在没有关于待聚类数据集先验知识的情况下，在不同的簇具有不同形状的数据集上，可能采用不同的结构元素来进行形态学操作，从而得到不同簇核心后，再进行集成得到一个簇核心，它在理论上效果应该更好。所以我们提出了基于数学形态学的聚类集成算法 CEOMM，它利用不同的结构元素的探针作用，对不同的结构元素探测出来的簇核心图进行集成，在集成所得到的簇核心基础上进行聚类。对于由不同形状的类构成的数据集，由于采用了不同的结构元素进行探测，理论上它比只使用一种结构元素进行探测更理想。而且在没有关于待聚类数据集中类的个数及形状大小方面的领域知识时，使用这种集成方法可能要比盲目使用一种结构元素进行探测所冒的风险更低。对每种结构元素的探测可以并行进行，所以它的时间复杂性并不会提高。当然如果集体中大多数的结构元素选择不合适，可能集成后的结果也不理想。实验表明这种算法确实能识别出形状复杂且形状各异的簇。

在我们的实验中没有涉及高维数据集，对于高维数据集，我们的设想是利用聚类集成思想，每次处理两个维度，然后对所得结果进行集成。这也将是我们下一步的工作。

在本文的算法中没有考虑噪声和异常点的问题，而实际上数学形态学具有很好的消除噪声影响以及发现异常点的功能。我们可以利用形态操作后得到簇核心来决定哪些点是噪声，或异常点。而异常点的检测在银行、证券、保险业、电信及网络安全等领域中都是非常重要的。

参 考 文 献

[1] Breen E J, Jones R, Talbot H. Mathematical morphology: A useful set of tools for image analysis [J]. Statistics and Computing, 2000, 10(2): 105-120

[2] Matheron F. Random Sets and Integral Geometry [M]. New York: John Wiley & Sons Inc, 1975

[3] Serra J. Image Analysis and Mathematical Morphology [M]. London: Academic Press, 1984

[4] Gonzalez R C, Woods R E. Digital Image Processing (2nd Edition) [M]. Prentice Hall, 2002

[5] Mcreynolds D P, Lowe D G. Geodesic saliency of watershed contours and hierarchical segmentation [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 1996, 8(12): 1174-1185

[6] Breen E J, Jones R, Talbot H. Mathematical morphology: A useful set of tools for image analysis [J]. Statistics and Computing, 2000, 10(2): 105-120

[7] Postaire J G, Zhang R D, Lecocq-Botte C. Clustering analysis by binary morphology [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 1993, 15(2): 170-180

[8] Starovoitov V. A clustering technique based on the distance transform [J]. Pattern Recognition Letters, 1996, 17(3): 231-239

[9] 邸凯昌，李德仁，李德毅. 从空间数据库发现聚类：一种基于数学形态学的算法 [J]. 中国图象图形学报，1998，3(3)：173-178

[10] Ferreira Costa J A, De Andrade Netto M L. Clustering of complex shaped data sets via Kohonen maps and mathematical morphology[C]// Proc. SPIE Data Mining and Knowledge Discovery: Theory, Tools, and Technology III. 2001: 16-27

[11] Guha S, Rastogi R, Shim K. Cure: An efficient clustering algorithm for large databases[C]// Proceedings of the ACM SIGMOD Conference on Management of Data. Seattle, WA, 1998: 73-84

[12] Ester M, Kriegel H-P, Sander J, et al. A density-based algorithm for discovering clusters in large spatial databases with noise[C]// Proceedings of the 2nd International Conference on Know-ledge Discovery and Data Mining (KDD-96). Portland, OR, 1996: 226-231

[13] Karyapis G, Han E H, Kumar V. CHAMELEON: A Hierarchical Clustering Algorithm Using Dynamic Modeling [J]. IEEE Computer, Special Issue on Data Analysis and Mining), 1999: 68-75

[14] Wu S, Chow T W S. Clustering of the self-organizing map using a clustering validity index based on inter-cluster and intra-cluster density [J]. Pattern Recognition, 2004, 37(2): 175-188

[15] 罗会兰，孔繁胜，扬小兵. 基于数学形态学的聚类分析 [J]. 模式识别与人工智能，2006，19(6)：727-733

[16] 唐常青，吕宏伯，黄铮. 数学形态学方法及其应用 [M]. 北京：科学出版社，1990

[17] 崔屹. 图象处理与分析一数学形态学方法及应用 [M]. 北京：科学出版社，2000

[18] 龚炜，石青云，程民德. 数字空间中的数学形态学一理论及应用 [M]. 北京：科学出版社，1997

[19] Theodoridis S, Koutroumbas K. 模式识别(第二版) [M]. 北京：电子工业出版社，2004

[20] Gonzalez R C, Woods R E. 数字图像处理(第二版) [M]. 北京：电子工业出版社，2003