基于离散指示矩阵优化的多视角图聚类方法

王　念，张　炜，崔智高*，苏延召，姜　柯，李爱华
（火箭军工程大学 陕西 西安，710025）

摘　要：为了解决当前多视角图聚类方法依赖于谱分解而导致计算量高且超参数调优过程繁琐的问题，基于离散指示矩阵优化的方式，提出了一种无超参数的多视角图聚类方法。为了避免谱分解的连续松弛，采用一种超集群策略来高效区分数据的离散集群关系。在优化层面，改进了传统的坐标下降方法，以降低计算复杂度以及实现离散指示矩阵的快速优化。在物体图像、人脸图像、文本数据、手写字体图像中进行了算法性能验证。结果表明：相比于最近常用的多视角图聚类方法，所提方法在聚类精度和运行效率方面具有明显优势。

关键词：多视角图聚类；离散指示矩阵；超集群策略；坐标下降；谱分解
中图分类号：TP391　　**文献标志码：**A　　**文章编号：**2097-2741(2024)03-051-009

A Graph-based Multi-view Clustering Method Based on Discrete Indicator Matrix Optimization

WANG Nian, ZHANG Wei, CUI Zhigao*, SU Yanzhao,
JIANG Ke, LI Aihua
(Rocket Force University of Engineering, Xi'an 710025, Shaanxi)

ABSTRACT: To solve problems of current graph-based multi-view clustering methods, such as big computational load and complex hyperparameter tuning process caused by overdependence on spectral decomposition, a high efficient graph-based multi-view clustering method with no hyperparameter was proposed by introducing discrete indicator matrix optimization. To avoid the continuous relaxation of spectral decomposition, a supercluster strategy was adopted to efficiently distinguish the discrete cluster relationships of data. For optimization, the traditional coordinate descent method was improved, which reduced the computational complexity, to realize the fast optimization of discrete indicator matrix. Additionally, the proposed method was verified in object images, face images, gene images and handwriting images. Results showed that the proposed method has obvious advantages in clustering accuracy and running efficiency compared with other graph-based multi-view clustering methods.

KEYWORDS: graph-based multi-view clustering; discrete indicator matrix; supercluster strategy; coordinate descent; spectral decomposition

收稿日期：2024-04-01
基金项目：陕西省自然科学基础研究计划（2023-JC-YB-501）
第一作者：王念（1996—），男，博士研究生，主要从事数据挖掘与图像增强研究。E-mail：1150712112@qq.com
***通信作者**：崔智高（1984—），男，副教授，主要从事智能视频监控研究。E-mail：cuizg@126.com

引用格式：王念，张炜，崔智高，等.基于离散指示矩阵优化的多视角图聚类方法[J].火箭军工程大学学报，2024，38（3）：51-59.

聚类是数据挖掘的基础，它不需要任何先验信息，通过数据自身的流形结构进行数据的集群划分，在地表信息勘测[1]、海上救援[2]、人脸识别[3]文本信息的智能检索[4]等领域广泛应用。由于数据获取的多样性，孕育了一个新的研究领域，即多视角聚类。多视角聚类通过多个视角提取输入数据的共性特性，以提高聚类精度。这基于一个共识，即多个渠道得出的共性知识更加可靠。多视角聚类在过去十年里得到了广泛研究[5]，可大致分为4类：共训练聚类[6]、多核聚类[7]、多视角子空间聚类[8]以及多视角图聚类。

多视角图聚类是利用多个视角数据的互补性，挖掘多个视角数据的共识图结构，以实现高精度的聚类结果[9]。多视角图聚类的关键在于如何利用多视角信息得到共识图，并输出离散的集群关系。由于图本身不反映集群关系，因此，需要使用离散的指示矩阵进行集群划分。由于离散的指示矩阵，不利于优化方案的设计，因此，通常需要将离散的指示矩阵松弛为连续的实值矩阵，再利用谱分解得到优化结果，最后使用额外的离散方法（通常为k-means，一种传统聚类方法）得到离散的集群关系。

根据谱分解使用方式的不同，可以将多视角图聚类方法分为3类。第1类方法[10]基于谱分解的常规流程，首先融合所有视角的信息得到共识相似度图，再对共识相似度图进行谱分解得到谱嵌入，最后通过k-means得到离散的聚类结果。这种方法实现简单，但由于k-means是一种随机性比较强的聚类方法，其聚类效果极度依赖于初始数据中心的给定位置，导致最后的聚类结果受到限制，并在反复的运行过程中波动[11]。第2类方法[12-13]使用正交谱旋转（也称为正交普鲁克分析）来离散化谱嵌入。尽管理想图的谱嵌入不是精确的离散指示矩阵，但其列向量张成了离散指示矩阵的一组正交基。根据这一性质，可以引入一个旋转矩阵，通过估计谱嵌入的旋转来直接恢复离散的指示矩阵，从而避免k-means的离散操作。这种方法的优势是能够获得绝对稳定的结果，并且其结果通常优于第1类方法。但是，这种方法引入了另一个优化问题来估计谱嵌入的最优旋转矩阵，这容易造成多个优化不匹配的问题。第3类方法[14-17]在图构建的过程中嵌入谱分解，通过正则的方式使学习的相似度矩阵为一个块对角矩阵，其对角块数量严格等于聚类数目。因此，学习的图被称为结构图，其连通分量直接提供离散的聚类指示。与第2类方法一样，结构图学习的方法能得到绝对稳定的聚类结果，并且由于学习的图不是预先固定的，而是作为变量在迭代优化中更新，这种方法比第2类方法更加有效。但是，由于需要利用超参数来平衡图学习和谱嵌入正则，其有效性严重依赖于超参数的取值。因此，需要手动设置超参数的取值范围和变化步长，利用试凑法搜寻最优结果，这一过程非常繁琐、耗时，不利于实际使用。

综上所述，尽管大量多视角图聚类工作已经提出，但当前的方法普遍依赖于谱分解，这种将离散指示矩阵松弛为连续实值后再恢复成离散形式的方法虽然便于优化过程的设计，但不可避免地造成了信息的损失，限制了聚类精度。此外，谱分解是特征值分解的一种特殊形式，其计算复杂度为$O(n^3)$，其中n为样本数量。当输入数据的规模较大时，算法的运行速度急速下降。

为了解决这些问题，本文摈弃了谱分解方法，研究了一种基于离散指示矩阵优化的多视角图聚类方法，利用图中的相似度关系，直接优化离散的聚类指示，指导集群的划分。在优化方面，改进了传统的坐标下降模式，并提出一种快速坐标下降方法，以降低计算复杂度。此外，本文方法不包含超参数，所以无需进行参数调优，并且能产生确定性的结果。

1 模型构建

1.1 图构建

多视角图聚类的第1步是如何构建每个视角的相似度矩阵（图）。常用的方法包括二进制（0-1）相似度、余弦相似度、高斯核相似度等。上述相似度度量方法通过固定距离函数确定任意2个数据点之间的相似性，这极大地限制了近邻范围确定的灵活性，并导致非稀疏的相似度矩阵。

一种更简单有效的方法是通过流形学习和稀疏表示来构建相似度矩阵[14]。流形学习的一个主要假设是：如果2个数据点的距离越近，它们就越有可能成为邻居。因此，令$\boldsymbol{X}^{(1)}, \boldsymbol{X}^{(2)}, \cdots, \boldsymbol{X}^{(v)}$表示具有$v$个视角的数据集（其中：$\boldsymbol{X}^{(m)} \in \mathbb{R}^{d_m \times n}$表示第$m$个视角的数据矩阵），概率邻居可以通过将相似度矩阵$\boldsymbol{S}$嵌入到距离函数（例如欧几里得距离$\| \boldsymbol{x}_i^{(m)} - \boldsymbol{x}_j^{(m)} \|_2^2$）来自动确定。

$$\begin{cases}\min\limits_{s_i^{(m)}} \sum\limits_{i,j=1}^{n} \| \boldsymbol{x}_i^{(m)} - \boldsymbol{x}_j^{(m)} \|_2^2 s_{ij}^{(m)} + \gamma (s_{ij}^{(m)})^2 \\ \text{s.t. } \mathbf{1}^{\mathrm{T}} \boldsymbol{s}_i^{(m)} = 1,\ 0 \leqslant s_{ij}^{(m)} \leqslant 1 \end{cases} \tag{1}$$

式中：γ为需要调节的超参数；$\boldsymbol{s}_i^{(m)} \in \mathbb{R}^{n\times 1}$表示第$j$个元素为$s_{ij}^{(m)}$的向量；$\mathbf{1}^{\mathrm{T}} \in \mathbb{R}^{n\times 1}$表示所有元素为1的向量。约束$\mathbf{1}^{\mathrm{T}}\boldsymbol{s}_i^{(m)} = 1$和$0 \leqslant s_{ij}^{(m)} \leqslant 1$，可使$s_{ij}^{(m)}$归一化为概率相似度。式（1）中的第1项通过欧式距离学习概率相似度矩阵$\boldsymbol{s}$，第2项则确保了$\boldsymbol{s}$的稀疏性。

令$\boldsymbol{d}_i^{(m)} \in \mathbb{R}^{n\times 1}$，其中第$j$个元素为$d_{ij}^{(m)} = \| \boldsymbol{x}_i^{(m)} - \boldsymbol{x}_j^{(m)} \|_2^2$，得到式（1）的向量形式为

$$\min_{\mathbf{1}^{\mathrm{T}}\boldsymbol{s}_i^{(m)} = 1,\ 0 \leqslant s_{ij}^{(m)} \leqslant 1} \| \boldsymbol{s}_i^{(m)} + \frac{1}{2\gamma} \boldsymbol{d}_i^{(m)} \|_2^2 \tag{2}$$

根据文献[14]，将$d_{i,1}^{(m)}, d_{i,2}^{(m)}, \cdots, d_{i,n}^{(m)}$从小到大排列，超参数$\gamma$可以取值为$\gamma = \frac{1}{n}\sum\limits_{i=1}^{n}(\frac{c}{2} d_{i,c+1}^{(m)} - \frac{1}{2}\sum\limits_{j=1}^{c} d_{ij}^{(m)})$。$\gamma$由近邻数目$c$自动调节，以避免超参数调优。利用Karush-Kuhn-Tucker条件，可以得到问题（2）的闭式解为

$$s_{ij}^{(m)} = \frac{d_{i,c+1}^{(m)} - d_{ij}^{(m)}}{c d_{i,c+1}^{(m)} - \sum\limits_{j=1}^{c} d_{ij}^{(m)}} \tag{3}$$

可以看到，这种构图方式有如下优势。

（1）式（3）只涉及简单的加减乘除运算，而传统构图方法（如高斯核函数）涉及更多类型的运算，如指数运算。

（2）学习到的相似度矩阵$\boldsymbol{S}$是自然稀疏的，因为式（3）自动筛选了最近的c个数据点构建c近邻图。对于图聚类来说，稀疏图在提升计算速度方面非常重要。

（3）s_{ij}具有尺度不变性。如果数据点$\boldsymbol{x}_1, \boldsymbol{x}_2, \cdots, \boldsymbol{x}_n$由任意标量$t$进行缩放，例如，将每一个数据点$\boldsymbol{x}_i$变为$t\boldsymbol{x}_i$，$d_{ij}$将变为$td_{ij}$，此时$s_{ij}$不会发生变化（这得益于式（3）中分子与分母的特殊形式）。然而，在这种情形下，传统构图方法会导致s_{ij}持续变化，使超参数难以调节。

（4）式（3）只含有1个超参数，即近邻数目c，它是一个整数，易于调节。大多数情况下，只需$5 \leqslant c \leqslant 10$就能得到合理的结果。这个性质非常重要，因为超参数调节在聚类任务中仍是一个难题。

1.2 超集群划分

由于构建的图只能表征所有样本之间的成对相似度，不具备提供离散聚类指示的能力，需要使用特定的图割方法来进行集群划分。聚类指示矩阵$\boldsymbol{F}$本身是离散的，其中每行只有1个元素为1（元素1的位置为当前样本的类别），其余元素为0。$\boldsymbol{F}$的离散性质导致难以设计对应的优化方案，因此，广泛使用的方法是谱聚类，它将$\boldsymbol{F}$松弛为连续实值，只保留其正交性质$\boldsymbol{F}^{\mathrm{T}}\boldsymbol{F} = \boldsymbol{I}$。通过这种方式，可以利用式（4）求得低维谱嵌入。

$$\min_{\boldsymbol{F}^{\mathrm{T}}\boldsymbol{F} = \boldsymbol{I}} \mathrm{Tr}(\boldsymbol{F}^{\mathrm{T}}\boldsymbol{L}\boldsymbol{F}) \tag{4}$$

式中：Tr ()表示矩阵的迹。

图的拉普拉斯矩阵$\boldsymbol{L} \in \mathbb{R}^{n\times n}$被定义为$\boldsymbol{L} = \boldsymbol{D} - \boldsymbol{S}$，度矩阵$\boldsymbol{D} \in \mathbb{R}^{n\times n}$是一个对角矩阵，其第$i$个对角元素$\boldsymbol{d}_j = \sum\limits_{j} s_{ij}$，即相似度矩阵$\boldsymbol{S}$第$i$行元素和，它代表了与数据点$i$连接的所有数据点（包括其自身）的权重和。式（4）可以通过瑞利商快速求解。因为求得的$\boldsymbol{F}$为连续的实值矩阵，需要额外的离散方法（通常为k-means）进行聚类。这种基于谱分解的方法造成了信息损失，并且计算量为$O(n^3)$，当样本数量n较大时，运行时间极速增加。但是，由于优化层面的限制，谱分解仍是当前广泛使用的方法。为了避免谱分解的使用，本节提出一种超集群划分策略，方便使用坐标下降的方法直接优化离散的聚类指示矩阵。

首先考虑单视角的情况。令Q为包含所有样本的数据集，Q_l表示第l个集群中所有数据点的集合，$\bar{Q}_l$为Q中Q_l的补集。根据图割的基本思想，应该最小化所有集合$(Q_1, \cdots, Q_k)$之间的相似度，其中k为需要的聚类数目。为了更新集合Q_l中的元素，可以将它的补集$\bar{Q}_l$看作超集群（其他所有集群的并集），最小化Q_l和$\bar{Q}_l$之间的相似度。因此，对应的相似度度量准则为

$$\mathrm{sim}(Q_l, \bar{Q}_l) = \frac{\sum\limits_{i\in Q_l}\sum\limits_{j\in \bar{Q}_l}(s_{i,j} + s_{j,i})}{\sum\limits_{i\in Q_l}\sum\limits_{j\in Q_l}(s_{i,j} + s_{j,i}) \sum\limits_{i\in \bar{Q}_l}\sum\limits_{j\in \bar{Q}_l}(s_{i,j} + s_{j,i})} \tag{5}$$

式中：分子$\sum\limits_{i\in Q_l}\sum\limits_{j\in \bar{Q}_l}(s_{i,j} + s_{j,i})$表示$Q_l$和$\bar{Q}_l$之间所有边的权重（相似度）和；分母中$\sum\limits_{i\in Q_l}\sum\limits_{j\in Q_l}(s_{i,j} + s_{j,i})$和$\sum\limits_{i\in \bar{Q}_l}\sum\limits_{j\in \bar{Q}_l}(s_{i,j} + s_{j,i})$分别表示$Q_l$和$\bar{Q}_l$中所有边的权重和。

容易看出，最小化式（4）通过使用图中的所

有权重，增强了 Q_l 和 $\bar{Q}_l$ 的内聚性，同时最大化它们之间的差异。

同时考虑所有集群的更新，进一步得到

$$\begin{cases} \min_{Q_l} \sum_{l=1}^{k} \dfrac{\sum\limits_{i\in Q_l}\sum\limits_{j\in \bar{Q}_l}(s_{i,j}+s_{j,i})}{\sum\limits_{i\in Q_l}\sum\limits_{j\in Q_l}(s_{i,j}+s_{j,i})\sum\limits_{i\in \bar{Q}_l}\sum\limits_{j\in \bar{Q}_l}(s_{i,j}+s_{j,i})} \\ \text{s.t.}\ \bigcup_{l=1}^{k} Q_l = Q,\ \forall m,\ n\in \mathbb{Z}_{[1,k]},\ Q_m \cap Q_n = \varnothing \end{cases} \tag{6}$$

式中：$m, n \in \mathbb{Z}_{[1,k]}$ 表示取值范围为[1, k]的整数；$\varnothing$ 表示空集。式（6）同时最小化所有情况下的式（5），协同更新所有 k 个集群中的元素。

基于式（6）的特殊形式，可以使用离散的聚类指示矩阵 $\boldsymbol{F}\in\mathbb{R}^{n\times k}$ 替代繁琐的集合表示。$\boldsymbol{F}$ 中的每一行元素为离散的聚类指示向量 $\boldsymbol{f}\in\mathbb{R}^{n\times 1}$。$\boldsymbol{f}$ 为二值向量，只有1个元素为1，其余全为0，1的位置表示该数据点的类别索引。基于此，问题（6）等价于

$$\min_{\boldsymbol{F}\in \mathrm{Ind}} \sum_{l=1}^{k} \frac{\boldsymbol{f}_l^{\mathrm{T}}\boldsymbol{L}\boldsymbol{f}_l / (\boldsymbol{f}_l^{\mathrm{T}}\boldsymbol{S}\boldsymbol{f}_l)}{\sum\limits_{i,j=1}^{n} s_{i,j} - 2\boldsymbol{f}_l^{\mathrm{T}}\boldsymbol{S}\boldsymbol{1} + \boldsymbol{f}_l^{\mathrm{T}}\boldsymbol{S}\boldsymbol{f}_l} \tag{7}$$

考虑所有视角，得到本文方法的目标函数为

$$\begin{cases} \min\limits_{\boldsymbol{F},\boldsymbol{\alpha}} \sum\limits_{m=1}^{v} \alpha^{(m)} \sum\limits_{l=1}^{k} \dfrac{\boldsymbol{f}_l^{\mathrm{T}}\boldsymbol{L}^{(m)}\boldsymbol{f}_l / (\boldsymbol{f}_l^{\mathrm{T}}\boldsymbol{S}^{(m)}\boldsymbol{f}_l)}{\sum\limits_{i,j=1}^{n} s_{i,j}^{(m)} - 2\boldsymbol{f}_l^{\mathrm{T}}\boldsymbol{S}^{(m)}\boldsymbol{1} + \boldsymbol{f}_l^{\mathrm{T}}\boldsymbol{S}^{(m)}\boldsymbol{f}_l} \\ \text{s.t.}\ \boldsymbol{F}\in \mathrm{Ind},\ \sum\limits_{m=1}^{v} \dfrac{1}{\alpha^{(m)}} = 1, \{\alpha^{(m)}\}_v \geqslant 0 \end{cases} \tag{8}$$

式中：$\boldsymbol{F}\in\mathrm{Ind}$ 表示 $\boldsymbol{F}$ 为离散的指示矩阵；$\{\alpha^{(m)}\}_v \geqslant 0$ 保证视角权值的非负性；$\sum\limits_{m=1}^{v}\dfrac{1}{\alpha^{(m)}} = 1$ 是尺度约束，保证 $\boldsymbol{F}$ 更新为最优解 $\boldsymbol{F}^*$ 时，第 m 个视角的最优权重 $(\alpha^{(m)})^*$ 与目标函数值负相关，即

$$(\alpha^{(m)})^* \propto 1 / \sum_{l=1}^{k} \frac{\boldsymbol{f}_l^{*\mathrm{T}}\boldsymbol{L}^{(m)}\boldsymbol{f}_l^* / (\boldsymbol{f}_l^{*\mathrm{T}}\boldsymbol{S}^{(m)}\boldsymbol{f}_l^*)}{\sum\limits_{i,j=1}^{n} s_{i,j}^{(m)} - 2\boldsymbol{f}_l^{*\mathrm{T}}\boldsymbol{S}^{(m)}\boldsymbol{1} + \boldsymbol{f}_l^{*\mathrm{T}}\boldsymbol{S}^{(m)}\boldsymbol{f}_l^*} \tag{9}$$

由式（9）可知，所采用的视角加权策略具有物理意义：如果第 m 个视角的目标函数值较大，则最优权重 $(\alpha^{(m)})^*$ 自动减小，降低第 m 个视角中相似度矩阵和拉普拉斯矩阵的重要性。

2 优化设计

本文方法有2个变量需要优化，即权重向量 $\boldsymbol{\alpha}$ 和离散聚类指标矩阵 $\boldsymbol{F}$。因此，需要随机初始化 $\boldsymbol{\alpha}$ 和 $\boldsymbol{F}$，并采用交替迭代策略实现参数的自动更新。

2.1 固定 $\boldsymbol{F}$，更新 $\boldsymbol{\alpha}$

固定 $\boldsymbol{F}$，$\sum\limits_{l=1}^{k} \dfrac{\boldsymbol{f}_l^{*\mathrm{T}}\boldsymbol{L}^{(m)}\boldsymbol{f}_l^* / (\boldsymbol{f}_l^{*\mathrm{T}}\boldsymbol{S}^{(m)}\boldsymbol{f}_l^*)}{\sum\limits_{i,j=1}^{n} s_{i,j}^{(m)} - 2\boldsymbol{f}_l^{*\mathrm{T}}\boldsymbol{S}^{(m)}\boldsymbol{1} + \boldsymbol{f}_l^{*\mathrm{T}}\boldsymbol{S}^{(m)}\boldsymbol{f}_l^*}$ 的值变为一个常数。通过用 $\eta^{(m)}$ 表示这一常数，式（8）可以改写为

$$\begin{cases} \min\limits_{\{\alpha^{(m)}\}_v} \sum\limits_{m=1}^{v} \alpha^{(m)}\eta^{(m)} \\ \text{s.t.}\ \sum\limits_{m=1}^{v} \dfrac{1}{\alpha^{(m)}} = 1,\ \alpha^{(m)} \geqslant 0 \end{cases} \tag{10}$$

由于约束 $\sum\limits_{m=1}^{v}\dfrac{1}{\alpha^{(m)}} = 1$，基于柯西-许瓦兹（Cauchy-Schwarz）不等式得到

$$\left(\sum_{m=1}^{v}\alpha^{(m)}\eta^{(m)}\right)\left(\sum_{m=1}^{v}\frac{1}{\alpha^{(m)}}\right) \geqslant \left(\sum_{m=1}^{v}\sqrt{\eta^{(m)}}\right)^2 \tag{11}$$

当且仅当 $\sqrt{\alpha^{(m)}\eta^{(m)}}$ 与 $\sqrt{\dfrac{1}{\alpha^{(m)}}}$ 线性相关，即 $\sqrt{\alpha^{(m)}\eta^{(m)}} = p\sqrt{\dfrac{1}{\alpha^{(m)}}}$，其中 p 为常数时，式（11）中的等式条件成立。结合 $\sum\limits_{m=1}^{v}\dfrac{1}{\alpha^{(m)}} = 1$，得到

$$\sum_{m=1}^{v}\sqrt{\alpha^{(m)}\eta^{(m)}} = \sum_{m=1}^{v} p\sqrt{\frac{1}{\alpha^{(m)}}} \Rightarrow$$

$$\sum_{m=1}^{v}\frac{1}{\alpha^{(m)}} = \sum_{m=1}^{v}\frac{\sqrt{\eta^{(m)}}}{p} \Rightarrow p = \sum_{m=1}^{v}\sqrt{\eta^{(m)}} \tag{12}$$

因此，权重向量 $\boldsymbol{\alpha}$ 可以由下式更新

$$\alpha^{(m)} = \left(\sum_{i=1}^{v}\sqrt{\eta^{(i)}} / \sqrt{\eta^{(m)}}\right) \tag{13}$$

2.2 固定 $\boldsymbol{\alpha}$，更新 $\boldsymbol{F}$

固定 $\boldsymbol{\alpha}$，式（8）可以改写为

$$\min_{\boldsymbol{F}\in \mathrm{Ind}} \sum_{l=1}^{k} \frac{\boldsymbol{f}_l^{\mathrm{T}}\hat{\boldsymbol{L}}\boldsymbol{f}_l / (\boldsymbol{f}_l^{\mathrm{T}}\hat{\boldsymbol{S}}\boldsymbol{f}_l)}{\sum\limits_{i,j=1}^{n}\hat{s}_{i,j} - 2\boldsymbol{f}_l^{\mathrm{T}}\hat{\boldsymbol{S}}\boldsymbol{1} + \boldsymbol{f}_l^{\mathrm{T}}\hat{\boldsymbol{S}}\boldsymbol{f}_l} \tag{14}$$

式中：$\hat{\boldsymbol{S}} = \dfrac{1}{v}\sum\limits_{m=1}^{v}\alpha^{(m)}\boldsymbol{S}^{(m)}$ 和 $\hat{\boldsymbol{L}} = \dfrac{1}{v}\sum\limits_{m=1}^{v}\alpha^{(m)}\boldsymbol{L}^{(m)}$ 分别表示加权后的相似度矩阵和拉普拉斯矩阵。

式（14）可以分为3部分：$\boldsymbol{f}_l^{\mathrm{T}}\hat{\boldsymbol{L}}\boldsymbol{f}_l$、$\boldsymbol{f}_l^{\mathrm{T}}\hat{\boldsymbol{S}}\boldsymbol{f}_l$ 和 $\sum\limits_{i,j=1}^{n}\hat{s}_{i,j} - 2\boldsymbol{f}_l^{\mathrm{T}}\hat{\boldsymbol{S}}\boldsymbol{1} + \boldsymbol{f}_l^{\mathrm{T}}\hat{\boldsymbol{S}}\boldsymbol{f}_l$，其中 $l\in 1, 2, \cdots, k$。初始化 $\boldsymbol{F}$ 之后，由于 l 的取值不同，上述3个部分分别派生出 k 个值，总计 $3k$ 个值。根据离散聚类指标的定义，$\boldsymbol{F}$ 是行无关的，每一行只有1个元素为1，其余为0。因此，传统坐标下降算法可以直接用于迭代求解问题（14），在此过程中，逐行将 $\boldsymbol{F}$ 更新为最优值。假设元素1在矩阵 $\boldsymbol{F}$ 第 i 行中的位置为 s，其中 $s\in 1, 2, \cdots, k$，当更新 $\boldsymbol{F}$ 第 i 行时，第 i 行

可以是{[1, 0, …, 0], …,[0, 0, …, 1]}的k种情况之一。为表述方便，引入指标矩阵$\boldsymbol{F}^{(s)}$来表述第i行中1的位置，$\boldsymbol{F}^{(s)}$取其中一种{$\boldsymbol{F}^{(1)}$, $\boldsymbol{F}^{(2)}$, …, $\boldsymbol{F}^{(k)}$}。请注意，$\boldsymbol{F}^{(s)}$与$\boldsymbol{F}$除第i行外，其余元素完全相同。为了更加直观地表示这种区别，图1展示了$\boldsymbol{F}^{(s)}$与$\boldsymbol{F}$的图形可视化结果。

因此，问题（14）的目标函数可改写为

$$\begin{cases}\min\limits_{\boldsymbol{F}^{(s)}} L(\boldsymbol{F}^{(s)})=\sum\limits_{l=1}^{k}\dfrac{\boldsymbol{f}_l^{(s)\mathrm{T}}\hat{\boldsymbol{L}}\boldsymbol{f}_l^{(s)}/(\boldsymbol{f}_l^{(s)\mathrm{T}}\hat{\boldsymbol{S}}\boldsymbol{f}_l^{(s)})}{\sum\limits_{i,j=1}^{n}\hat{s}_{i,j}-2\boldsymbol{f}_l^{(s)\mathrm{T}}\hat{\boldsymbol{S}}\boldsymbol{1}+\boldsymbol{f}_l^{(s)\mathrm{T}}\hat{\boldsymbol{S}}\boldsymbol{f}_l^{(s)}} \\ \text{s.t. } \boldsymbol{F}^{(s)}\in\{\boldsymbol{F}^{(1)},\boldsymbol{F}^{(2)},\cdots,\boldsymbol{F}^{(k)}\}\end{cases} \tag{15}$$

可以看出，计算一种$\boldsymbol{F}^{(s)}$的复杂度为$O(3kn^2)$，k种情况的复杂度$O(3k^2n^2)$。此外，需要重复上述步骤n次，以更新$\boldsymbol{F}$的所有行，则总的复杂度为$O(3k^2n^2)$，计算量非常大。

为了降低复杂度，引入$\boldsymbol{F}^{(0)}$。如图1所示，在更新$\boldsymbol{F}$第i行时，$\boldsymbol{F}^{(0)}$与$\boldsymbol{F}$除第i行外，其余元素完全相同，其中$\boldsymbol{F}^{(0)}$的第i行全为零。根据$\boldsymbol{F}^{(s)}$、$\boldsymbol{F}^{(0)}$和$\boldsymbol{F}$的定义，从矩阵的列方向看，它们只在第s列（第i个数据的标签）有区别，因此，无需迭代第k列而只更新第s列，得到式（15）的等价问题为

$$\begin{cases}\min\limits_{\boldsymbol{F}^{(s)}}\mathcal{L}(\boldsymbol{F}^{(s)})=\dfrac{\boldsymbol{f}_l^{(s)\mathrm{T}}\hat{\boldsymbol{L}}\boldsymbol{f}_l^{(s)}/(\boldsymbol{f}_l^{(s)\mathrm{T}}\hat{\boldsymbol{S}}\boldsymbol{f}_l^{(s)})}{\sum\limits_{i,j=1}^{n}\hat{s}_{i,j}-2\boldsymbol{f}_l^{(s)\mathrm{T}}\hat{\boldsymbol{S}}\boldsymbol{1}+\boldsymbol{f}_l^{(s)\mathrm{T}}\hat{\boldsymbol{S}}\boldsymbol{f}_l^{(s)}}- \\ \qquad\dfrac{\boldsymbol{f}_l^{(0)\mathrm{T}}\hat{\boldsymbol{L}}\boldsymbol{f}_l^{(0)}/(\boldsymbol{f}_l^{(0)\mathrm{T}}\hat{\boldsymbol{S}}\boldsymbol{f}_l^{(0)})}{\sum\limits_{i,j=1}^{n}\hat{s}_{i,j}-2\boldsymbol{f}_l^{(0)\mathrm{T}}\hat{\boldsymbol{S}}\boldsymbol{1}+\boldsymbol{f}_l^{(0)\mathrm{T}}\hat{\boldsymbol{S}}\boldsymbol{f}_l^{(0)}} \\ \text{s.t. } \boldsymbol{F}^{(s)}\in\{\boldsymbol{F}^{(1)},\boldsymbol{F}^{(2)},\cdots,\boldsymbol{F}^{(k)}\}\end{cases} \tag{16}$$

可以看出，总的复杂度被降为$O(3kn^3)$，但仍然需要较高的运算量。为了进一步降低计算量，需要通过已知的$3k$个值$\boldsymbol{f}_l^{\mathrm{T}}\hat{\boldsymbol{L}}\boldsymbol{f}_l$、$\boldsymbol{f}_l^{\mathrm{T}}\hat{\boldsymbol{S}}\boldsymbol{f}_l$和$\sum\limits_{i,j=1}^{n}\hat{s}_{i,j}-2\boldsymbol{f}_l^{\mathrm{T}}\hat{\boldsymbol{S}}\boldsymbol{1}+\boldsymbol{f}_l^{\mathrm{T}}\hat{\boldsymbol{S}}\boldsymbol{f}_l$（$l\in 1, 2, \cdots, k$）来代替式（16）中的部分计算。

具体地，假设1位于初始化$\boldsymbol{F}$的第q列。当更新$\boldsymbol{F}$的第i行时，由于问题（16）的解取决于元素1是否在$\boldsymbol{F}^{(s)}$和$\boldsymbol{F}$的相同位置，即s是否等于q，所以通过引入一个列向量$\boldsymbol{\delta}$作为变换因子（其中：第i个元素为1，其余元素为0），得到以下分组结果：

1）当$s=q$时，可以得到$\boldsymbol{f}_s^{(s)}=\boldsymbol{f}_s$和$\boldsymbol{f}_s^{(0)}=\boldsymbol{f}_s-\boldsymbol{\delta}$。将它们代入$\mathcal{L}(\boldsymbol{F}^{(s)})$的4个部分，得到

$$\mathcal{L}(\boldsymbol{F}^{(s)})=\frac{\boldsymbol{f}_s^{\mathrm{T}}\hat{\boldsymbol{L}}\boldsymbol{f}_s/(\boldsymbol{f}_s^{\mathrm{T}}\hat{\boldsymbol{S}}\boldsymbol{f}_s)}{\sum\limits_{i,j=1}^{n}\hat{s}_{i,j}-2\boldsymbol{f}_s^{\mathrm{T}}\hat{\boldsymbol{S}}\boldsymbol{1}+\boldsymbol{f}_s^{\mathrm{T}}\hat{\boldsymbol{S}}\boldsymbol{f}_s}- \frac{(\boldsymbol{f}_s^{\mathrm{T}}\hat{\boldsymbol{L}}\boldsymbol{f}_s-2\boldsymbol{f}_s^{\mathrm{T}}\hat{\boldsymbol{L}}_{:,i}+\hat{L}_{ii})/(\boldsymbol{f}_s^{T}\hat{\boldsymbol{S}}\boldsymbol{f}_s-2\boldsymbol{f}_s^{T}\hat{\boldsymbol{S}}_{:,i}+\hat{S}_{ii})}{\sum\limits_{i,j=1}^{n}\hat{s}_{i,j}-2\boldsymbol{f}_s^{\mathrm{T}}\hat{\boldsymbol{S}}\boldsymbol{1}+2\boldsymbol{1}^{\mathrm{T}}\hat{\boldsymbol{S}}_{:,i}+\boldsymbol{f}_s^{\mathrm{T}}\hat{\boldsymbol{S}}\boldsymbol{f}_s-2\boldsymbol{f}_s^{\mathrm{T}}\hat{\boldsymbol{S}}_{:,i}+\hat{\boldsymbol{S}}_{ii}} \tag{17}$$

2）当$s\neq q$时，可以得到$\boldsymbol{f}_s^{(s)}=\boldsymbol{f}_s+\boldsymbol{\delta}$和$\boldsymbol{f}_s^{(0)}=\boldsymbol{f}_s$。将它们代入$\mathcal{L}(\boldsymbol{F}^{(s)})$的4个部分，得到

$$\mathcal{L}(\mathrm{F}^{(s)})=-\frac{\boldsymbol{f}_s^{\mathrm{T}}\hat{\boldsymbol{L}}\boldsymbol{f}_s/(\boldsymbol{f}_s^{\mathrm{T}}\hat{\boldsymbol{S}}\boldsymbol{f}_s)}{\sum\limits_{i,j=1}^{n}\hat{s}_{i,j}-2\boldsymbol{f}_s^{\mathrm{T}}\hat{\boldsymbol{S}}\boldsymbol{1}-\boldsymbol{f}_s^{\mathrm{T}}\hat{\boldsymbol{S}}\boldsymbol{f}_s}+ \frac{(\boldsymbol{f}_s^{\mathrm{T}}\hat{\boldsymbol{L}}\boldsymbol{f}_s+2\boldsymbol{f}_s^{\mathrm{T}}\hat{\boldsymbol{L}}_{:,i}+\hat{L}_{ii})/(\boldsymbol{f}_s^{\mathrm{T}}\hat{\boldsymbol{S}}\boldsymbol{f}_s+2\boldsymbol{f}_s^{\mathrm{T}}\hat{\boldsymbol{S}}_{:,i}+\hat{\boldsymbol{S}}_{ii})}{\sum\limits_{i,j=1}^{n}\hat{s}_{i,j}-2\boldsymbol{f}_s^{\mathrm{T}}\hat{\boldsymbol{S}}\boldsymbol{1}-2\boldsymbol{1}^{\mathrm{T}}\hat{\boldsymbol{S}}_{:,i}+\boldsymbol{f}_s^{\mathrm{T}}\hat{\boldsymbol{S}}\boldsymbol{f}_s+2\boldsymbol{f}_s^{\mathrm{T}}\hat{\boldsymbol{S}}_{:,i}+\hat{\boldsymbol{S}}_{ii}} \tag{18}$$

在式（17）和（18）中，除了$\boldsymbol{f}_s^{\mathrm{T}}\hat{\boldsymbol{L}}_{:,i}$和$\boldsymbol{f}_s^{\mathrm{T}}\hat{s}_{:,i}$之外，其余所有元素都是已知的，总的时间复杂度被降低为$O(3kn^2)$。

最后，需要用优化后的$\boldsymbol{F}^{(s)}$更新$\boldsymbol{F}$，即更新其中每行元素1的位置，直到收敛。收敛条件设置为$\|\boldsymbol{F}-\boldsymbol{F}^{(s)}\|_F^2\leqslant 10^{-9}$（其中$\|\cdot\|_F$为Frobenius范数），用于衡量矩阵中所有元素的差异，判断$\boldsymbol{F}^{(s)}$是否已经不再需要更新，即是否等于$\boldsymbol{F}$。

3　实验与讨论

3.1　对比算法

选择了8种具有代表性的方法，与本文算法的性能进行定量比较。这些方法包括AMGL（Auto-weighted Multiple Graph Learning）[9]，AWP（Ad-

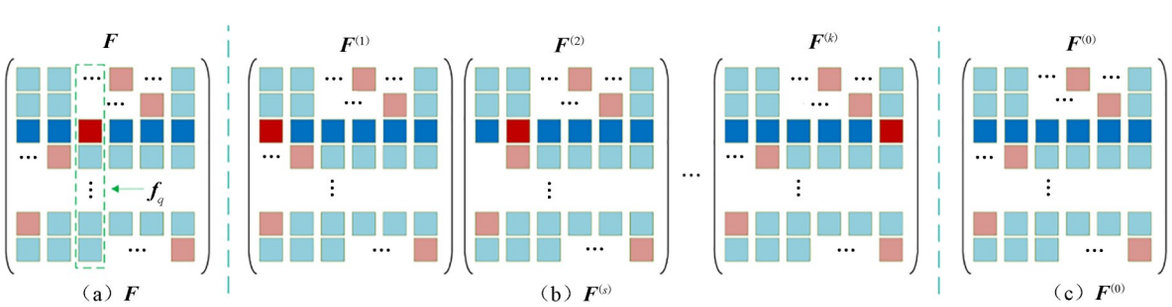

图1　$\boldsymbol{F}$、$\boldsymbol{F}^{(s)}$和$\boldsymbol{F}^{(0)}$的图形可视化解释

Fig. 1　Graphic visual interpretation for $\boldsymbol{F}$, $\boldsymbol{F}^{(s)}$, and $\boldsymbol{F}^{(0)}$

aptively Weighted Procrustes)[11], MCGC (Multiview Consensus Graph Clustering)[17], MV-GL[13] (Multi-view Clustering with Graph Learning), CDMGC[16] (Consistent and Divergent Multi-view Graphy Clustering), GMC (Graph-based Multi-view Clustering)[14], ASMV[15] (Adaptive Structrual Multi-view Clustering), MCMLE[10] (Multi-view Spectral Clustering via Multi Laplacian Embedding)。这些方法的源程序都已在网上公开。对于所有方法，构图时近邻数量都设置为10，每种方法的超参数调整为原文建议的最优值。

所有实验在一台Windows 11计算机上进行，CPU为2.5 GHz Intel i9-12900H，RAM为32 GB，运行平台为MATLAB 2020 b。

3.2 数据集介绍

实验使用的测试数据集包括：物体图像数据集COIL20、MSRCV1和YaleA、文本数据集3Sources、人脸数据集ORL和ALDI、手写数字数据集HW2和Hdigit。所有数据集规格如表1所示。

3.3 定量结果

为了保证实验的公平性，将每种算法在原文建议的最佳参数设置下运行5次，并报告平均结果。采用准确度、归一化互信息指标来评估聚类性能，结果如表2和表3所示，算法运行时间如表4所示。

由表2和表3可以看出：本文方法在大多数基准数据集的测试中得到了最优结果（加粗数字），只有在YaleA和Hdigit基准测试中获得了次优的聚类结果；对于这8个基准的平均准确度和归一化互信息，本文方法远远超过其他方法，与次优结果相比，分别将准确度和归一化互信息提高了6.94%和5.07%。结果充分证明了本文方法对不同类型数据集具有良好的聚类效果。

表1 数据集参数

Tab. 1 Parameters of data sets

数据集	样本数	视角数	聚类数	特征1	特征2	特征3	特征4	特征5	特征6
COIL20	1 440	3	20	1 024	3 304	6 750	—	—	—
MSRCV1	210	6	7	1 302	48	512	100	256	210
3Sources	169	3	6	3 560	3 631	3 068	—	—	—
HW2	2 000	2	10	784	256	—	—	—	—
YaleA	165	3	15	9	512	50	—	—	—
ORL	400	4	40	512	59	864	254	—	—
Hdigit	10 000	2	10	784	256	—	—	—	—
ALOI	10 800	4	100	77	13	64	125	—	—

表2 不同方法的准确度对比

Tab. 2 Comparison of accuracies

方法	准确度								均值
	COIL20	MSRCV1	3Sources	HW2	YaleA	ORL	Hdigit	ALOI	
AMGL	10.06	20.95	67.51	68.93	66.06	66.50	53.96	48.07	50.26
AWP	85.76	84.76	69.23	89.05	**78.69**	79.75	74.28	75.00	79.57
MCGC	69.79	82.38	31.36	48.40	64.24	69.00	58.40	60.72	60.54
MVGL	75.21	76.19	32.54	97.95	43.03	55.25	99.46	46.25	65.74
CDMGC	86.25	87.61	34.91	98.80	64.24	70.00	OM	OM	73.64
GMC	79.10	89.52	69.23	99.40	67.88	83.75	**99.81**	57.05	80.72
ASMV	79.24	39.04	33.73	74.20	46.67	75.50	75.19	45.55	58.64
MCMLE	78.19	78.57	71.01	69.75	74.55	75.50	98.92	OM	78.07
本文方法	**90.56**	**90.48**	**78.11**	**99.60**	76.36	**88.50**	99.79	**77.89**	**87.66**

注：粗体表示最优值。

表3 不同方法的归一化互信息结果对比

Tab. 3 Comparison of normalized mutual information

方法	归一化互信息								均值
	COIL20	MSRCV1	3Sources	HW2	YaleA	ORL	Hdigit	ALOI	
AMGL*	4.89	3.92	57.68	84.79	72.46	86.71	36.97	70.52	52.24
AWP	90.38	74.18	63.13	73.81	79.17	89.83	79.06	82.49	79.01
MCGC*	80.08	70.88	9.52	52.82	70.94	82.41	47.34	65.09	59.89
MVGL	83.75	70.72	7.11	95.55	46.52	71.16	98.32	66.57	67.46
CDMGC*	92.49	79.16	6.07	97.23	68.59	86.13	OM	OM	71.61
GMC	91.89	81.63	54.80	98.53	70.41	92.60	**99.39**	73.50	82.84
ASMV	86.81	30.97	8.96	84.39	54.86	86.90	89.32	67.67	63.74
MCMLE	86.42	66.19	68.47	61.87	73.25	87.35	96.93	OM	77.21
本文方法	**93.17**	**82.20**	**68.49**	**98.94**	**83.07**	**94.50**	99.35	**83.55**	**87.91**

表4 不同方法的运行时间结果对比

Tab. 4 Comparison of time costs

方法	COIL20	MSRCV1	3Sources	HW2	YaleA	ORL	Hdigit	ALOI
AMGL*	6.249 3	1.332 8	0.986 3	14.635 9	0.816 3	4.929 3	621.463 1	732.164 3
AWP	0.819 9	0.359 1	0.832 8	1.862 6	0.812 6	1.215 6	536.856 4	668.426 4
MCGC*	15.566 5	2.486 0	1.563 6	31.663 5	1.482 3	7.831 8	＞100 0	＞100 0
MVGL	32.618 9	3.192 6	2.628 6	54.659 3	2.389 6	9.713 6	＞100 0	＞100 0
CDMGC*	12.459 3	1.243 6	1.072 6	20.933 2	1.105 3	4.616 8	OM	OM
GMC	9.199 3	1.782 6	1.453 6	15.619 3	1.435 2	2.339 3	＞100 0	＞100 0
ASMV	35.593 6	3.618 9	2.236 9	52.916 3	2.147 2	6.523 6	＞100 0	＞100 0
MCMLE	26.153 9	2.299 3	1.983 2	48.492 3	1.681 3	4.624 6	＞100 0	OM
本文方法	**0.682 6**	**0.221 5**	**0.124 6**	**0.793 2**	**0.114 7**	**0.385 6**	**4.186 8**	**5.191 6**

表2～3中，AMGL、MCGC和CDMGC聚类结果不稳定。随着不同运行而波动，主要原因是这几种方法依赖于k-means的后处理和参数调优，这引入了随机性。相比之下，本文方法能够得到确定性的聚类结果。

由表4可以看到，本文方法以非常快的运行速度输出聚类结果。对于超过10 000个数据样本的Hdigit和ALOI数据集，本文方法能在10 s内结束迭代并得到结果，而其他方法普遍需1 000 s以上的时间。更重要的是，CDMGC方法和MCMLE方法因为较高的存储负担而导致“内存不足（Out of Memory）”错误（表中以“OM”标记）。

3.4 分析讨论

1）时间复杂度

本文方法的计算复杂度由2个优化过程组成：通过式（12）更新$\boldsymbol{\alpha}$的复杂度为$O(kn)$，通过求解问题（15）更新$\boldsymbol{F}$的成本为$O(kn^2)$。因为$n \gg k$，参数$\boldsymbol{\alpha}$的复杂度可以忽略不计。假设总的迭代次数为t，本文方法总的复杂度为$O(tkn^2)$，远低于谱分解方法和传统坐标下降方法的复杂度$O(tkn^3)$。

2）收敛性

图2为本文方法在所有数据集的收敛曲线。

由图2可以看到：本文方法的收敛速度很快，在前几次迭代中显著降低目标函数值；更重要的是，本文方法的收敛速度对样本数量不敏感，在不同数据集中的收敛速度没有显著变化，在10回合内稳定收敛，远低于其他方法（通常需要100次以上的迭代）。这一优势进一步保证了本文方法的高效性。

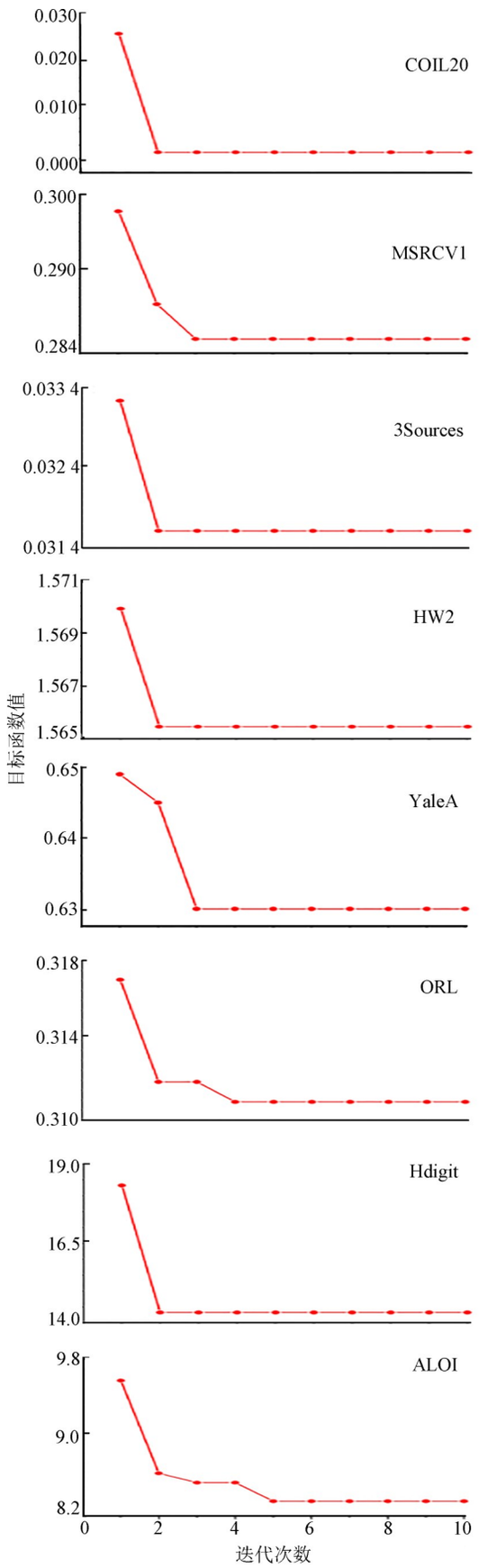

图 2 本文方法的收敛性曲线

Fig. 2 Convergence curves of the proposed method

4 结论

本文提出了一种全新的多视角图聚类方法，该方法巧妙地避免了谱分解，直接对离散的聚类指示矩阵进行优化，极大地提高了聚类的准确度和效率。此外，本文方法不包括超参数，不需要参数调优，便于实际应用。详细的理论推导和实验研究证明了本文方法的收敛性和效率。针对8种先进算法的对比实验表明本文方法具有极高的聚类准确度。在未来，我们将关注更多的图融合策略，以提高其性能，并利用一些先验信息或标记数据，研究半监督聚类方法。

参考文献：

[1] CHEN X H，ZHANG Y S，FENG X X，et al. Spectral-spatial superpixel anchor graph-based clustering for hyperspectral imagery[J]. IEEE Geoscience and Remote Sensing Letters，2023，20：5507405.

[2] TU B，WANG Z，OUYANG H T，et al. Hyperspectral anomaly detection using the spectral-spatial graph[J]. IEEE Transactions on Geoscience and Remote Sensing，2022，60：5542814.

[3] CHO Y，KIM W，HONG S，et al. Part-based pseudo label refinement for unsupervised person re-identification[C]//2022 IEEE / CVF Conference on Computer Vision and Pattern Recognition (CVPR). Piscataway，NJ，USA：IEEE，2022：7298-7308.

[4] KEYVANPOUR M，SERPUSH F. ESLMT：A new clustering method for biomedical document retrieval[J]. Biomedical Engineering，2019，64(6)：729-741.

[5] ZHANG Y，YANG Q. A survey on multi-task learning[J]. IEEE Transactions on Knowledge and Data Engineering，2022，34(12)：5586-5609.

[6] APPICE A，MALERBA D. A co-training strategy for multiple view clustering in process mining [J]. IEEE Transactions on Services Computing，2016，9(6)：832-845.

[7] YAO Y Q，LI Y，JIANG B B，et al. Multiple kernel k-means clustering by selecting representative kernels[J]. IEEE Transactions on Neural Networks and Learning Systems，2021，32(11)：4983-4996.

[8] ZHANG C Q，FU H Z，HU Q H，et al. Generalized latent multi-view subspace clustering[J].

IEEE Transactions on Pattern Analysis and Machine Intelligence，2020，42(1)：86-99.

[9] NIE F P，LI J，LI X L. Parameter-free auto-weighted multiple graph learning：A framework for multiview clustering and semi-supervised classification[C]//Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence. New York，USA：ACM，2016：1881-1887.

[10] ZHONG G，PUN C M. Improved normalized cut for multi-view clustering[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence，2022，44(12)：10244-10251.

[11] NIE F P，TIAN L，LI X L. Multiview clustering via adaptively weighted procrustes[C]//Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. New York，USA：ACM，2018：2022-2030.

[12] CHEN J，MAO H，PENG D Z，et al. Multiview clustering by consensus spectral rotation fusion [J]. IEEE Transactions on Image Processing，2023，32：5153-5166.

[13] ZHAN K，ZHANG C Q，GUAN J P，et al. Graph learning for multiview clustering[J]. IEEE Transactions on Cybernetics，2018，48(10)：2887-2895.

[14] WANG H，YANG Y，LIU B. GMC：Graph-based multi-view clustering[J]. IEEE Transactions on Knowledge and Data Engineering，2020，32(6)：1116－1129.

[15] ZHAN K，CHANG X J，GUAN J P，et al. Adaptive structure discovery for multimedia analysis using multiple features[J]. IEEE Transactions on Cybernetics，2019，49(5)：1826-1834.

[16] HUANG S D，TSANG I W，XU Z L，et al. Measuring diversity in graph learning：A unified framework for structured multi-view clustering[J]. IEEE Transactions on Knowledge and Data Engineering，2022，34(12)：5869-5883.

[17] ZHAN K，NIE F P，WANG J，et al. Multiview consensus graph clustering[J]. IEEE Transactions on Image Processing，2019，28(3)：1261-1270.