Neural Processing Letters (2023) 55:6067–6087
https://doi.org/10.1007/s11063-022-11127-7

Robust Dual-Graph Regularized Deep Matrix Factorization for Multi-view Clustering

Zhenqiu Shu[1] · Bin Li[1] · Cong Hu[2] · Zhengtao Yu[1] · Xiao-Jun Wu[2]

Accepted: 11 December 2022 / Published online: 22 February 2023

Abstract
The matrix factorization approaches have been widely applied for multi-view clustering since they can effectively explore complementary information contained in the multi-view data. However, some prior knowledge hidden in multi-view data cannot be fully exploited in existing matrix factorization based multi-view learning approaches. In this paper, we present a robust dual-graph regularized deep matrix factorization (RDDMF) approach for multi-view clustering. Specifically, it integrates the dual-graph regularizers and the sparse constraints into the deep matrix factorization framework. Therefore, the proposed RDDMF approach discovers the geometric structures of both the data and the feature space by adding the dual graph regularization term into deep matrix factorization in each layer. Meanwhile, the sparse constraints are imposed on the coefficient matrix of each layer to improve the robustness of our model. Besides, we design an efficient optimization strategy of the proposed model and give its convergence rate. Numerous experiments on four well-known datasets show our proposed RDDMF approach is superior to several state-of-the-art approaches in multi-view clustering.

Keywords Multi-view clustering · Deep matrix factorization · Dual-graph regularization · Sparse constraint

1 Introduction

In many applications, various descriptors, such as LBP, Color, HOG, etc, are used to depict an image. Consequently, the same object can be described from many different perspectives. A certain aspect of an object can be called a view. The data obtained by revealing different attributes of the same object is called multi-view data [1]. In the past few decades,

Zhenqiu Shu and Bin Li have contributed equally to this work.

✉ Zhenqiu Shu
shuzhenqiur@163.com

1 Faculty of Information Engineering and Automation, Kunming University of Science and Technology, Kunming 650500, Yunnan, China

2 Jiangsu Provincial Engineering Laboratory of Pattern Recognition and Computational Intelligence, Jiangnan University, Wuxi 214000, Jiangsu, China

many studies have focused on signal-view clustering [2–6]. Generally, the features extracted from different views can represent the object more comprehensively because different views contain generous supplementary information. Therefore, it is a foundation topic on how to improve the performance of multi-view learning algorithms by exploring and fusing multi-view features. Recently, multi-view learning is successfully applied to solve many practical problems, such as face recognition [7], image classification [8], natural language processing [9], disease diagnosis [10], etc. As a well-known topic in multi-view learning, multi-view clustering aims to adaptively divide the data samples into different groups using the feature information collected from the multi-view data [11, 12].

Extensive efforts from researchers have been devoted to multi-view clustering in the past few decades. Kumar et al. [1] imposed a co-regularization constraint on different views and then performed spectral clustering. Gao et al. [13] proposed to perform clustering on the subspace representation of each view while guaranteeing consistency of different views. Liu et al. [14] used the joint matrix factorization (MF) method to learn a consensus representation from the clustering solution of different views. Kamalika et al. [15] employed the CCA technology to map each view to a subspace and then performed clustering on this subspace. Wang et al. [16] proposed to jointly learn latent representations with clustering structures by using the MF-based multi-view clustering framework. Zhang et al. [17] encoded the image descriptors into a compact Hamming space and then optimized the multi-view clustering structure in this space. Li et al. [18] proposed to learn latent subspace by using a partial multi-view clustering approach. To preserve the local geometric structure embedded into different views, Zong et al. [19] presented a multi-manifold regularized NMF for multi-view clustering. The above-mentioned models, however, usually are a single-layer structure and thus cannot effectively extract the structural information hidden in multi-view data [20]. Motived by the deep learning structure, Zhao et al. [21] explored the semantic information of the multi-view data using deep semi-NMF and achieved promising clustering performance in multi-view clustering. Huang et al. [22] proposed an adaptive weighted multi-view clustering approach via deep matrix factorization. Wei et al. [23] employed deep matrix factorization technology to decompose the multi-view data into the representation subspaces. The aforementioned approaches, however, cannot take full advantage of the prior knowledge hidden in multi-view data. Therefore, they cannot adequately describe the complex multi-view data, and thus their clustering performances are limited in some cases.

In this paper, we propose a robust dual-graph regularized deep matrix factorization (RDDMF) for multi-view clustering. Specifically, we take full advantage of the deep matrix factorization structure to capture the semantic information of multi-view data. The graph regularization technology is used to capture the dual geometric structure, and the robustness of the proposed algorithm can be improved by adding the sparse constraints. Extensive experiments show that our proposed method achieves better performance compared with other state-of-the-art competitors.

The main contributions of this work can be summarized as follows:

(1) The proposed RDDMF approach adopts the deep semi-NMF structure to explore the latent semantic information contained in the multi-view data.
(2) In each decomposition, the proposed method not only explores the intrinsic geometric structure of multi-view data in the feature space and the data space by constructing a dual graph regularizer, but also imposes the sparse constraints on the coefficient vectors to improve the robustness in clustering

(3) Moreover, the iterative updating rules are developed to solve the proposed model. The experimental results on four multi-view datasets validate the superiority of the proposed RDDMF approach.

The rest of the paper is organized as follows. In Sect. 2, we review some related works. Section 3 introduces the proposed approach in detail. Section 4 shows the experimental evaluations and analysis. Section 5 gives the conclusion of this work.

2 Related Work

In this section, we introduce some matrix factorization approaches commonly used in multi-view clustering in detail.

2.1 NMF and Semi-NMF

Let $X = [x_1, x_2, \cdots, x_n,] \in R^{m \times n}$ be a nonnegative data matrix, where m denotes the dimension of the sample x_i and n denotes the number of samples. In NMF, the original data matrix X is decomposed into the basis matrix $U = [u_{ij}] \in R^{m \times c}$ and the coefficient matrix $V = [v_{ij}] \in R^{c \times n}$. The NMF model is formulated as the following minimization problem:

$$O_{NMF} = \sum_{i=1}^{m} \sum_{j=1}^{n} (X_{ij} - (UV)_{ij})^2 = \|X - UV\|_F^2$$
$$s.t. U \geq 0, V \geq 0 \tag{1}$$

where $\|.\|_F$ denotes the Frobenius norm of the matrix. Lee et al. [24] proposed a multiplicative iterative optimization algorithm for solving Eq. (1). Therefore, the updating rules of Eq. (1) are given as follows:

$$u_{ij} \leftarrow u_{ij} \frac{(XV^T)_{ij}}{(UVV^T)_{ij}} \tag{2}$$

$$v_{ij} \leftarrow v_{ij} \frac{(U^T X)_{ij}}{(U^T UV)_{ij}} \tag{3}$$

To solve the issue of negative elements in real data, Ding et al. [25] further developed a Semi-NMF approach for clustering. It only requires that the coefficient matrix is nonnegative. Thus, Semi-NMF seeks to decompose the original data matrix $X \in R^{m \times n}$ into two low-dimensional data matrices $U \in R^{m \times c}$ and $V \in R^{c \times n}$, where $V \geq 0$. Therefore, its model is formulated as follows:

$$\min_{U, V \geq 0} \|X - UV\|_F^2, \quad s.t. V \geq 0 \tag{4}$$

Thus, we also adopt similar iterative optimization strategy to solve Eq. (4).

2.2 Deep Semi-NMF

The research of deep learning theory have shown that deep network structures can comprehensively capture the feature information hidden in data. Trigeorgis et al. [20] introduced a Deep Semi-NMF method to exploit the hierarchical information of complex data. Suppose

there are R layers in the deep framework, and l_r denotes the layer size of the r-th layer. In the first layer, original data matrix X is decomposed into the basis matrix $U_1 \in R^{m\times l_1}$ and the coefficient matrix $V_1 \in R^{l_1\times n}$ ($V_1 \geq 0$). In the second layer, the coefficient matrix V_1 is used as the input matrix and performed the same operation as the first layer (i.e. $V_1 = U_2V_2$, where $U_2 \in R^{l_1\times l_2}$, $V_2 \in R^{l_2\times n}$). After performing deep matrix decomposition, the final coefficient matrix and basis matrix can be obtained. Therefore, this decomposition process of Deep Semi-NMF can be summarized as follows:

$$\begin{aligned} X &\approx U_1V_1^+, \\ V_1 &\approx U_2V_2^+, \\ &\vdots \\ V_{R-1} &\approx U_RV_R^+ \end{aligned} \tag{5}$$

where U_r and V_r denote the basis matrix and the coefficient matrix in r-th layer, respectively. The objective function of Deep Semi-NMF can be further written as follows:

$$X \approx U_1U_2\cdots U_RV_R^+ \tag{6}$$

The Deep Semi-NMF algorithm aims at minimizing the following problem:

$$\begin{aligned} O_F &= \|X - U_1\cdots U_r\cdots U_RV_R\|_F^2, \\ &s.t. V_r \geq 0, r = 1, \cdots, R \end{aligned} \tag{7}$$

Similarly, the model (7) can be solved by the following updating rules:

$$U_r = (\Psi_{r-1}^T\Psi_{r-1})^{-1}\Psi_{r-1}^T X\tilde{V}_r^T(\tilde{V}_r\tilde{V}_r^T)^{-1} \tag{8}$$

$$V_r = V_r\sqrt{\frac{[\Psi_r^TX]^+ + [\Psi_r^TX]^-V_r}{\Psi_r^TX]^- + [\Psi_r^TX]^+V_r}} \tag{9}$$

where $\Psi_r = U_1U_2\cdots U_r$ and $\Psi_0 = 1$. $\widetilde{V}_r$ denotes the latent feature by reconstructing the coefficient matrix $V_r(\widetilde{V}_r = U_{r+1}V_{r+1})$. $[M]^+$ means a matrix with all negative elements to 0 and $[M]^-$ means a matrix with all positive elements to 0. Their definition can be given as follows:

$$\forall i, j, [M]_{i,j}^+ = \frac{|M_{ij}|+M_{ij}}{2}, [M]_{i,j}^- = \frac{|M_{ij}|-M_{ij}}{2} \tag{10}$$

2.3 DMVC

Inspired by deep semi-NMF, Zhao et al. [21] proposed the DMVC method for multi-view clustering, which effectively explores the intrinsic structure of data and seeks to learn a common representation of multi-view data. The objective function of DMVC is formulated as follows:

$$\begin{aligned} \min_{U_l^{(s)}, V_l^{(s)}, V_l, \alpha^{(s)}} &= \sum_{s=1}^{S}(\alpha^{(s)})^\gamma\left(\left\|X^{(s)} - U_1^{(s)}U_2^{(s)}\ldots U_R^{(s)}V_R\right\|_F^2 + \beta tr(V_RL^{(s)}V_R{}^T)\right) \\ s.t.\quad & V_r^{(s)} \geq 0, V_R \geq 0, \sum_{s=1}^{S}\alpha^{(s)} = 1, \alpha^{(s)} \geq 0 \end{aligned} \tag{11}$$

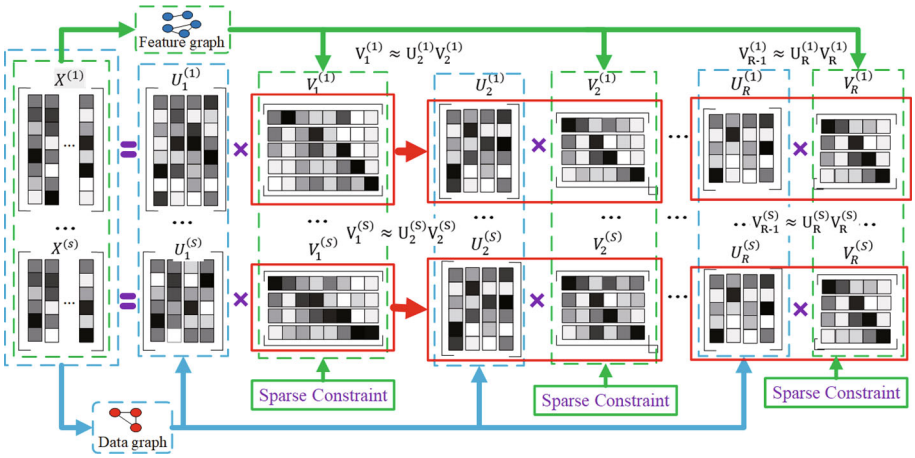

Fig. 1 The framework of the proposed RDDMF approach

where V_R denotes the common representation, which enables multi-view data to share the same representation after R-layer decomposition. $L^{(s)}$ denotes the Laplacian of the graph for the s-th view, which is used to preserve the geometric structure of multi-view data.

3 The Proposed Methods

In this section, we present the proposed RDDMF approach in multi-view clustering in detail.

3.1 Motivation

Traditional matrix factorization based multi-view clustering approaches aim at learning the common respresentation from the multi-view data, and then perform the clustering on the common respesentation to obtain the final labels. However, they cannot take full advantage of the prior knowledge embedding in multi-view data. Therefore, the clustering performances are seriously affected in many applications. In this paper, we propose a novel approach, termed robust dual-graph regularized deep matrix factorization (RDDMF), for multi-view clustering. Figure 1 shows the framework of our proposed RDDMF approach. Specifically, our proposed approach explores the semantic information of multi-view data via the deep matrix factorization. In addition, we construct a dual graph regularization term to model the manifold structure of both the feature and the data space, and then integrate it into matrix decomposition in each layer which is different from existing methods that only use graph regularization for the last layer. Meanwhile, the sparse constraint is imposed on the coefficient matrix of each layer and thus the robustness of the proposed approach can be significantly improved in multi-view clustering. Therefore, our proposed RDDMF approach shows more representation ability in dealing with complex multi-view data.

3.2 The Objective Function

We introduce the notations used in this paper. Denote the multi-view data with S views as $X = \{X^{(1)}, X^{(2)}, \cdots, X^{(S)}\}$, where the s-th view denotes $X^{(s)} = [x_1^{(s)}, x_2^{(s)}, \cdots, x_n^{(s)}] \in$

$R^{m_s \times n}$. m_s denotes the feature dimension of the s-th view and n is the number of training samples. $U_r^{(s)}$ and $V_r^{(s)}$ represent the basis matrix and the coefficient matrix of the r-th layer of the s-th view, respectively.

In the data space, we construct a p-nearest graph to model the geometric structure of the representation coefficients in each layer. Therefore, the affinity matrix of the data graph is given as follows:

$$W_{ij}^{(V^{(s)})} = \begin{cases} 1, & if \quad x_j^{(s)} \in N_p\left(x_i^{(s)}\right), \\ 0, & otherwise. \end{cases} \tag{12}$$

where $N_p(x_i^{(s)})$ represents the nearest neighbor set of $x_i^{(s)}$. The low-dimensional representation of the original data in data space can be expressed as $V_r^{(s)} = \left[(v_r^{(s)})_1, (v_r^{(s)})_2, \cdots, (v_r^{(s)})_n\right] \in R^{l_r \times n}$. $A^{V^{(s)}}$ denotes a diagonal matrix and its Laplacian matrix is $L^{V^{(s)}} = A^{V^{(s)}} - W^{V^{(s)}}$. Therefore, the graph regularization term of each layer in the data space can be given as follows:

$$\begin{aligned} R_V^{(s)} &= \frac{1}{2}\sum_{i,j=1}^{n} \left\| (v_r^{(s)})_i - (v_r^{(s)})_j \right\|^2 W^{V^{(s)}} \\ &= \sum_{i=1}^{n} (v_r^{(s)})_i (v_r^{(s)})_i{}^T A_{ii}^{V^{(s)}} - \sum_{i,j=1}^{n} (v_r^{(s)})_i (v_r^{(s)})_i{}^T W_{ij}^{V^{(s)}} \\ &= Tr(V_r^{(s)} A^{V^{(s)}} V^{(s)T}) - Tr(V_r^{(s)} W^{V^{(s)}} V_r^{(s)T}) \\ &= Tr(V_r^{(s)} L^{V^{(s)}} V_r^{(s)T}) \end{aligned} \tag{13}$$

Similarly, we construct a p-nearest neighbour graph in the feature space. The affinity matrix $W_{ij}^{U^{(s)}}$ of the feature graph is given as follows:

$$W_{ij}^{(U^{(s)})} = \begin{cases} 1, & if \quad x_j^{(s)T} \in N_p\left(x_i^{(s)T}\right), \\ 0, & otherwise. \end{cases} \tag{14}$$

where $N_p(x_i^{(s)T})$ represents the p-nearest neighbor set of $x_i^{(s)T}$. The low-dimensional representation of the original data in the feature space can be expressed as $U_r^{(s)} = \left[(u_r^{(s)})_1, (u_r^{(s)})_2, \cdots, (u_r^{(s)})_{m_s}\right] \in R^{m_s \times l_r}$. $A^{U^{(s)}}$ is a diagonal matrix and the Laplacian matrix is $L^{U^{(s)}} = A^{U^{(s)}} - W^{U^{(s)}}$. The graph regularization term of each layer in the feature space can be given as follows:

$$\begin{aligned} R_U^{(s)} &= \frac{1}{2}\sum_{i,j=1}^{m_s} \left\| (u_r^{(s)})_i - (u_r^{(s)})_j \right\|^2 W^{U^{(s)}} \\ &= \sum_{i=1}^{m_s} (u_r^{(s)})_i (u_r^{(s)})_i{}^T A_{ii}^{U^{(s)}} - \sum_{i,j=1}^{m_s} (u_r^{(s)})_i (u_r^{(s)})_i{}^T W_{ij}^{U^{(s)}} \\ &= Tr(U_r^{(s)T} A^{U^{(s)}} U_r^{(s)}) - Tr(U_r^{(s)T} W^{U^{(s)}} U_r^{(s)}) \\ &= Tr(U_r^{(s)T} L^{U^{(s)}} U_r^{(s)}) \end{aligned} \tag{15}$$

Therefore, we formulate our RDDMF model as follows:

$$\begin{aligned} O_F = & \sum_{s=1}^{S} (\alpha^{(s)})^{\gamma} \left\| X^{(s)} - \Psi_{r-1}^{(s)} U_r^{(s)} V_r^{(s)} \right\|_F^2 + \sum_{s=1}^{S} \mu \left\| V_r^{(s)} \right\|_{2,\frac{1}{2}}^{\frac{1}{2}} \\ & + \sum_{s=1}^{S} (\lambda Tr(V_r^{(s)} L^{V^{(s)}} V_r^{(s)^T}) + \beta Tr(U_r^{(s)^T} L^{U^{(s)}} U_r^{(s)})), \\ & s.t. \quad U_r^{(s)} \geq 0, V_r^{(s)} \geq 0 \end{aligned} \tag{16}$$

where λ, β and μ denote the regularization parameters. $\alpha^{(s)}$ is the weight coefficient of the s-th views, and γ denotes a tuning parameter that controls the weights distribution between different views. $L^{V^{(s)}}$ and $L^{U^{(s)}}$ denote the Laplacian matrix of both the data graph and the feature graph in rth layer, respectively.

We can further rewrite Eq. (16) into the following form:

$$\begin{aligned} O_F = & \sum_{s=1}^{S} (\alpha^{(s)})^{\gamma} \left\| X^{(s)} - \Psi_{r-1}^{(s)} U_r^{(s)} V_r^{(s)} \right\|_F^2 \\ & + \sum_{s=1}^{S} 4\mu Tr(V_r^{(s)} M_r^{(s)} V_r^{(s)^T}) \\ & + \sum_{s=1}^{S} (\lambda Tr(V_r^{(s)} L^{V^{(s)}} V_r^{(s)^T}) + \beta Tr(U_r^{(s)^T} L^{U^{(s)}} U_r^{(s)})), \\ & s.t. \quad U_r^{(s)} \geq 0, V_r^{(s)} \geq 0, \sum_{s=1}^{S} \alpha^{(s)} = 1, \alpha^{(s)} \geq 0 \end{aligned} \tag{17}$$

where $M = [m_{ij}] \in R^{n \times n}$ is the diagonal matrix. According to [26], the i-th diagonal element of the matrix M can be calculated as follows:

$$m_{ii} = \frac{1}{4 \left\| v_i \right\|_2^{\frac{3}{2}}} \tag{18}$$

3.3 Optimization

It is obvious that the model (17) is non-convex and thus cannot obtain its global optimization solution. Therefore, we update one variable while fixing other variables. Therefore, the objective function (17) can be further formulated as follows:

$$\begin{aligned} L = & \sum_{s=1}^{S} (\alpha^{(s)})^{\gamma} (Tr(X^{(s)} X^{(s)^T}) - 2Tr(\Psi_{r-1}^{(s)} U_R^{(s)} V_R^{(s)} X^{(s)^T}) \\ & + Tr(\Psi_{r-1}^{(s)} U_R^{(s)} V_R^{(s)} V_R^{(s)^T} U_R^{(s)^T} \Psi_{r-1}^{(s)^T})) \\ & + \sum_{s=1}^{S} 4\mu Tr(V_r^{(s)} M_r^{(s)} V_r^{(s)^T}) + \sum_{s=1}^{S} (\lambda Tr(V_r^{(s)} L^{V^{(s)}} V_r^{(s)^T}) \\ & + \beta Tr(U_r^{(s)^T} L^{U^{(s)}} U_r^{(s)})), \end{aligned}$$

$$s.t.\quad U_r^{(s)} \geq 0, V_r^{(s)} \geq 0, \sum_{s=1}^{S} \alpha^{(s)} = 1, \alpha^{(s)} \geq 0 \tag{19}$$

(1) Update $\mathbf{U_r^{(s)}}$: we obtain the Lagrange function of $U_r^{(s)}$ as follows:

$$\begin{aligned} L(U_r^{(s)}) = & -2Tr(\Psi_{r-1}^{(s)} U_R^{(s)} V_R^{(s)} X^{(s)^T}) \\ & + Tr(\Psi_{r-1}^{(s)} U_R^{(s)} V_R^{(s)} V_R^{(s)^T} U_R^{(s)^T} \Psi_{r-1}^{(s)^T}) \\ & + \beta Tr(U_r^{(s)^T} L^{U^{(s)}} U_r^{(s)}). \end{aligned} \tag{20}$$

By taking the partial derivation of $L\left(U_r^{(s)}\right) w.r.t.\, U_r^{(s)}$, we have

$$\begin{aligned} \frac{\partial L(U_r^{(s)})}{\partial U_r^{(s)}} = & -2\Psi_{r-1}^T X^{(s)} (\tilde{V}_r^{(s)})^T + 2\beta L^{U^{(s)}} U_r^{(s)} \\ & + 2\Psi_{r-1}^T \Psi_{r-1} U_r^{(s)} \tilde{V}_r^{(s)} (\tilde{V}_r^{(s)})^T \end{aligned} \tag{21}$$

Then by setting $\frac{\partial L\left(U_r^{(s)}\right)}{\partial U_r^{(s)}}$ to zero, the updating rule for the variable $U_r^{(s)}$ is given as follows:

$$U_r^{(s)} = U_r^{(s)} \times \frac{\Psi_{r-1}^T X^{(s)} (\tilde{V}_r^{(s)})^T + \beta W^{U^{(s)}} U_r^{(s)}}{\Psi_{r-1}^T \Psi_{r-1} U_r^{(s)} \tilde{V}_r^{(s)} (\tilde{V}_r^{(s)})^T + \beta A^{U^{(s)}} U_r^{(s)}} \tag{22}$$

(2) Update $\mathbf{V_r^{(s)}}$: The Lagrange function of $V_r^{(s)}$ is given as follows:

$$\begin{aligned} L(V_r^{(s)}) = & -2Tr(\Psi_{r-1}^{(s)} U_R^{(s)} V_R^{(s)} (X^{(s)})^T) \\ & + Tr(\Psi_{r-1}^{(s)} U_R^{(s)} V_R^{(s)} (V_R^{(s)})^T (U_R^{(s)})^T (\Psi_{r-1}^{(s)})^T) \\ & + 4\mu Tr(V_r^{(s)} M_r^{(s)} (V_r^{(s)})^T) \\ & + \lambda Tr(V_r^{(s)} L^{V^{(s)}} (V_r^{(s)})^T). \end{aligned} \tag{23}$$

By calculating the partial derivation of $L\left(V_r^{(s)}\right) w.r.t.\, V_r^{(s)}$, we have

$$\begin{aligned} \frac{\partial L(V_r^{(s)})}{\partial V_r^{(s)}} = & -2(\Psi_r^{(s)})^T X^{(s)} + 2(\Psi_r^{(s)})^T \Psi_r^{(s)} V_r^{(s)} \\ & + 2\lambda V_r^{(s)} L^{V^{(s)}} + 8\mu (V_r^{(s)})^T M_r^{(s)} \end{aligned} \tag{24}$$

Then we set $\frac{\partial L\left(V_r^{(s)}\right)}{\partial U_r^{(s)}}$ to zero and get the updating rule for the variable $V_r^{(s)}$ as follows:

$$V_r^{(s)} = V_r^{(s)} \sqrt{\frac{[\Psi_r^T X^{(s)}]^+ + [\Psi_r^T \Psi_r V_r^{(s)}]^- + \lambda V_r^{(s)} W^{V^{(s)}}}{[\Psi_r^T X^{(s)}]^- + [\Psi_r^T \Psi_r V_r^{(s)}]^+ + \lambda V_r^{(s)} A^{V^{(s)}} + 4\mu (V_r^{(s)})^T M_r^{(s)}}} \tag{25}$$

(3) Update $^{(\mathbf{s})}$: For convenience, we denote $\varphi^{(s)}$ as the following form:

$$\begin{aligned} \varphi^{(s)} = & \left\| X^{(s)} - \Psi^{(s)} U_r^{(s)} V_r^{(s)} \right\|_F^2 + 4\mu Tr(V_r^{(s)} M_r^{(s)} V_r^{(s)^T}) \\ & + \lambda Tr(V_r^{(s)} L^{V^{(s)}} V_r^{(s)^T}) + \beta Tr(U_r^{(s)^T} L^{U^{(s)}} U_r^{(s)}) \end{aligned} \tag{26}$$

Therefore, the objective function (17) is further simplified to the following problem:

$$O_F = \sum_{s=1}^{S} (\alpha^{(s)})^{\gamma} \varphi^{(s)}, s.t. \sum_{s=1}^{S} \alpha^{(s)} = 1, \alpha^{(s)} \geq 0 \tag{27}$$

Then Eq. (27) is reformulated as follows:

$$L(\alpha^{(s)}) = \sum_{s=1}^{S} (\alpha^{(s)})^{\gamma} \varphi^{(s)} + \theta \left(1 - \sum_{s=1}^{S} \alpha^{(s)}\right) \tag{28}$$

where θ is the Lagrange multiplier. By taking the partial derivation of Eq. (28) for $\alpha^{(s)}$, and then setting $\frac{\partial L(\alpha^{(s)})}{\partial \alpha^{(s)}}$ to zero, we have

$$\alpha^{(s)} = \frac{(\gamma \varphi^{(s)})^{\frac{1}{1-\gamma}}}{\sum_{s=1}^{S} (\gamma \varphi^{(s)})^{\frac{1}{1-\gamma}}} \tag{29}$$

The total optimization procedure of the proposed RDDMF approach is summarized in Algorithm 1.

Algorithm 1 The RDDMF approach

Require: Multi-view data $X^{(s)}$, parameters γ, λ, β and μ, layers, the number of nearest neighbors k.
Ensure: Basis matrix $U_r^{(s)}$ and coefficient matrix $V_r^{(s)}$ in each layer.

Initialize:
for all layer **do**
  $(U_r^{(s)}, V_r^{(s)}) \leftarrow$ SemiNMF $(V_{r-1}^{(s)}, layers)$;
  $W^{V^{(s)}} \leftarrow$ data graph construction on $x_i^{(s)}$;
  $W^{U^{(s)}} \leftarrow$ feature graph construction on ${x_i^{(s)}}^T$;
  Initialize the weight factor $\alpha^{(s)} = \frac{1}{S}$ for each view;
end for
while not converged **do**
  for all layer **do**
    Update $U_r^{(s)}$ according to Eq. (22);
    Update $V_r^{(s)}$ according to Eq. (25);
    Update $\alpha^{(s)}$ according to Eq. (29);
  end for
end while

Finally, we get the consensus representation V by the following formula:

$$V = \sum_{s=1}^{S} \alpha^{(s)} V_R^{(s)} \tag{30}$$

3.4 Computational Complexity

The procedure of the proposed approach mainly includes pre-training and fine-tuning. R is the number of layers, and the sizes of all layers are set to l. S denotes the number of views. The original feature dimensions for all the views are denoted d. In the first stage, the computational complexity of the pre-training process using the Semi-NMF method is

$O(SRT(dnl + nl^2 + ld^2 + ln^2 + dn^2 + nd^2))$. Normally, $l << d$, thus $\tau_{pre-traing} = O(SRT(ld^2 + dn^2 + nd^2))$. Similarly, we get the computational complexity of the second stage $\tau_{fine-tuning} = O(SRT(ld^2+2dl+d^2l+2n^2l))$, where T is the number of iterations. To sum up, the overall computational complexity of the proposed RDSMF approach is $\tau_{tatal} = \tau_{pre-training} + \tau_{fine-tuning} = O(SRT(2ld^2 + ld(2 + d) + 2ln^2 + dn^2 + dn^2))$.

4 Experiments

In this section, we conducted some experiments to validate the effectiveness of the proposed RDDMF approach on four benchmark datasets (i.e. HandWritten, Extended Yale B, Notting-Hill and COIL20). To make a fair comparison, some state-of-the-art multi-view clustering approaches are used as comparison algorithms. Several widely-used metrics, such as clustering accuracy (ACC), normalized mutual information (NMI), Adjusted Rand Index (ARI), F-score, Precision and Recall, were adopted to evaluate the performances of different approaches in multi-view clustering tasks.

4.1 Evaluation Metrics

(1) ACC: It can be calculated by the following formula:

$$ACC = \frac{\sum_1^n \partial(g_i, map(l_i))}{n} \tag{31}$$

where l_i is the output label of the model, g_i is the true label of the sample and n is the number of samples. $map(*)$ is the optimal mapping function. The delta function $\partial(\cdot) = 1$ if $g_i = map(l_i)$. Otherwise, $\partial(\cdot) = 0$.

(2) NMI: The mutual information (MI) can be calculated as follows:

$$MI(L, G) = \sum_{l_i \in C, g_i \in G} p(l_i, g_i) log_2^{\frac{p(l_i, g_i)}{p(l_i) \cdot p(g_i)}}, \tag{32}$$

where $p(\cdot)$ is the probability and $p(\cdot, \cdot)$ is the joint probability. Then NMI is given as follows:

$$NMI(L, G) = \frac{MI(L, G)}{max(H(L), H(G))} \tag{33}$$

where $H(\cdot)$ is the entropy function. L is the label set obtained by the model and G is the true label set of the sample. As you can see, the more similar the two categories are, the greater the value of NMI is.

(3) ARI: First, RI can be defined as follows:

$$RI = \frac{TP + TN}{TP + FP + TN + FN} = \frac{TP + TN}{C_N^2} \tag{34}$$

where C_N^2 indicates how many combinations of any two samples belong to the same category, and N denotes the number of samples. RI cannot guarantee that the RI value of randomly divided clustering results is close to 0, and thus we introduce the adjusted RI, namely ARI. It can be obtained as follows:

$$ARI = \frac{RI - E(RI)}{max(RI) - E(RI)} \tag{35}$$

where $E(\cdot)$ denotes the expected index.

(4) Precision, Recall and F-score: We first introduce four concepts: *a*) TP: It indicates that similar samples are assigned to the same category; *b*) TN: It means that different types of samples are assigned to different categories; *c*) FP: It indicates that different types of samples are classified into the same category; *d*) FN: This means that similar samples are classified into different categories. Therefore, Precision can be defined as follows:

$$Precision = \frac{TP}{TP + FP} \tag{36}$$

It is obvious that Precision is the percentage of true positives in the retrieved results. Therefore, Recall can be defined as follows:

$$Recall = \frac{TP}{TP + FN} \tag{37}$$

And F-score can be defined as follows:

$$F - score = (1 + \tau^2)\frac{Precision \cdot Recall}{\tau^2(Precision + Recall)} \tag{38}$$

where the parameter τ is used to balance the weight of Precision and Recall.

4.2 Datasets

The four benckmark multi-view datasets are introduced as follows:

(1) **HandWritten** is a handwritten digital image dataset consisting of 2000 samples belonging to 10 categories. Different features extracted from the picture are used as different views, they are 76-D Fourier coefficients (FOU), 216-D contour correlation (FAC), 64-D Karhunen-love coefficients (KAR), 64-D Karhunen-love coefficients (KAR), 240-D Pixel Mean (PIX), 47-D Zernike Moment (ZER), and 6-D Morphology (MOR) in 2×3 Window Features .

(2) **Extended YaleB** consists of 64 face images of 38 individuals under different lighting conditions and postures. Here, a total of 640 images of the first 10 subjects are selected as the experimental sub dataset. Similarly, the density, LBP and Gabor features of each sample are described by a 2500-dimensional, 3304-dimensional, 6750- dimensional vector, respectively.

(3) **Notting-Hill** is generated from the movie 'Notting Hill', which is a widely used video face benchmark dataset. It mainly involves 4660 faces of 5 major casts in 76 tracks [27]. Similarly, we can represent the density, LBP and Gabor of each sample by a 2000-dimensional, a 3304-dimensional, and a 6750-dimensional vector, respectively.

(4) **COIL20** consists of 1440 images from 20 objects including toy ducks, and mugs. Each image is taken at 5-degree intervals of an object spinning on a turntable. Similar to the previous datasets, we extracted three different features (i.e., intensity, LBP and Gabor) as three different descriptions to construct a multi-view dataset.

Overall, Table 1 summarizes the four benchmark multi-view datasets in detail.

4.3 Baseline Methods

To fairly evaluate the effectiveness of the proposed approach in multi-view clustering, we compare it with the following multi-view baseline approaches:

Table 1 Statistics of datasets

Dataset	Size	Dimensions of views	View	Cluster
HandWritten	2000	76, 216, 64, 240, 47, 6	6	10
Extended Yale B	640	2500, 3304, 6750	3	38
Notting-Hill	4660	2000, 3304, 6750	3	5
COIL20	1440	1024, 3304, 6750	3	20

(1) BestSV: This method performs standard spectral clustering in each view and reports the best performance as the final result [28].
(2) ConFea: It performs standard spectral clustering on matrices concatenating all original data features.
(3) ConPCA: This approach uses Principal Component Analysis (PCA) method to project the original multi-view data matrix after concatenating all features into a low-dimensional subspace for clustering.
(4) Co-Reg (SPC) [1]: It co-regularizes the clustering hypotheses to enforce data points with the same attribute in each view to the same category.
(5) Co-Training (SPC) [1]: It uses the idea of co-training to bootstrap the clustering of different views using spectral embedding from one another.
(6) MultiNMF [14]: This method applies NMF to project multi-view data into a common latent subspace.
(7) NaMSC [29]: It combines the representations learned by using smooth representation clustering (SRC) and feeds in each view data to the spectral clustering.
(8) DiMSC [30]: This approach learns all the different subspace representations by introducing a diversity constraint term.
(9) DMVC [21]: This method adopts the deep matrix factorization method to learn the multi-view data.
(10) PLCMF [16]: It performs clustering on each view to generate the pseudo-labels and then guides the collective matrix factorization procedure.
(11) NMFCC [31]: It designs an NMF model with co-orthogonal constraints to avoid performance degradation in multi-view clustering due to the orthogonality of the learned basis matrix and the internal vectors of the representation matrix.

4.4 Clustering Results

In our experiments, we repeated the multi-view clustering approaches ten times and reported their mean and standard deviations. The best results in the tables are highlighted in bold.

Tables 2 and 3 show the clustering results of the proposed RDDMF method and other competitors on HandWritten and Extended YaleB datasets, respectively. From Tables 2 and 3, it can be observed that our RDDMF method outperforms other multi-view clustering methods. Specifically, ConPCA achieves the best performance among all comparison algorithms on the HandWritten dataset. Compared with ConPCA, our RDDMF method improves around 1.1%, 1.3%, 8.0%, 1.0%, 1.8% and 1.9% over ConPCA in terms of ACC, NMI, ARI, F-score, Precision and Recall, respectively. Similarly, from Table 3, it is clear to see that the DMVC method outperforms other baseline methods on the Extended YaleB dataset. Nevertheless, compared with DMVC, our proposed RDDMF approach improves about 2.1% on ACC, 4.7% on NMI, 2.4% on ARI, 1.8% on F-score, 3.4% on precision and 2.7% on recall. The main

Table 2 Clustering performance on HandWritten dataset

	ACC	NMI	ARI	F-score	Precision	Recall
BestSV	0.537±0.001	0.476±0.003	0.523±0.014	0.573±0.009	0.563±0.008	0.581±0.011
ConFea	0.605±0.024	0.584±0.031	0.612±0.041	0.596±0.022	0.593±0.034	0.599±0.028
ConPCA	0.957±0.012	0.908±0.017	0.908±0.021	0.917±0.018	0.917±0.021	0.918±0.019
Co-Reg	0.551±0.024	0.479±0.041	0.543±0.028	0.468±0.031	0.579±0.027	0.623±0.037
Co-Train	0.503±0.019	0.384±0.027	0.494±0.036	0.536±0.024	0.509±0.023	0.567±0.022
MultiNMF	0.841±0.037	0.753±0.052	0.841±0.029	0.726±0.031	0.724±0.041	0.738±0.029
NaMSC	0.553±0.015	0.555±0.007	0.423±0.012	0.484±0.014	0.454±0.021	0.519±0.011
DiMSC	0.493±0.003	0.355±0.012	0.213±0.005	0.301±0.011	0.266±0.014	0.347±0.012
DMVC	0.522±0.011	0.502±0.014	0.541±0.016	0.586±0.022	0.616±0.016	0.625±0.017
PLCMF	0.922±0.046	0.864±0.029	0.922±0.034	0.867±0.027	0.869±0.036	0.895±0.022
NMFCC	0.865±0.016	0.869±0.020	0.864±0.016	0.757±0.024	0.752±0.018	0.763±0.022
Ours	0.968±0.025	0.921±0.031	0.918±0.034	0.936±0.021	0.935±0.031	0.937±0.024

Table 3 Clustering performance on extended YaleB dataset

	ACC	NMI	ARI	F-score	Precision	Recall
BestSV	0.366±0.059	0.360±0.016	0.225±0.018	0.303±0.011	0.296±0.010	0.310±0.012
ConFea	0.224±0.012	0.147±0.005	0.064±0.003	0.159±0.002	0.155±0.002	0.162±0.002
ConPCA	0.232±0.005	0.152±0.003	0.069±0.002	0.161±0.002	0.158±0.001	0.164±0.002
Co-Reg	0.224±0.000	0.151±0.001	0.066±0.001	0.160±0.000	0.157±0.001	0.162±0.000
Co-Train	0.186±0.001	0.302±0.007	0.043±0.001	0.140±0.001	0.137±0.001	0.143±0.002
MultiNMF	0.428±0.002	0.377±0.006	0.231±0.001	0.329±0.001	0.298±0.001	0.372±0.002
NaMSC	0.581±0.013	0.594±0.004	0.380±0.002	0.446±0.004	0.411±0.002	0.486±0.001
DiMSC	0.615±0.003	0.635±0.002	0.453±0.000	0.504±0.006	0.481±0.002	0.534±0.001
DMVC	0.763±0.001	0.649±0.002	0.512±0.002	0.564±0.001	0.525±0.001	0.610±0.001
PLCMF	0.330±0.008	0.298±0.008	0.349±0.002	0.272±0.002	0.233±0.006	0.326±0.001
NMFCC	0.333±0.025	0.294±0.031	0.145±0.0015	0.238±0.013	0.217±0.013	0.260±0.018
Ours	0.784±0.002	0.696±0.005	0.536±0.005	0.582±0.002	0.559±0.004	0.637±0.003

reason is that DMVC adopts the deep matrix factorization method to explore the semantic information of multi-view data. However, it only considers the manifold geometric structure of the coefficient vectors of the last layer in the data space. Compared with DMVC, our proposed RDDMF approach constructs a dual graph regularization term in each layer to model the intrinsic manifold geometric embedded in the feature and the data space, and uses the sparse constraints to improve the robustness of complex multi-view data representation, simultaneously.

Table 4 lists the results of our RDDMF approach on the Notting-Hill dataset. Different with above-mentioned two datasets, the lighting conditions in the Notting-Hill dataset vary drastically with the movie scene. As shown in Table 4, DiMSC and DMVC achieve the two best results among the comparison algorithms in terms of all metrics. Especially, compared with DMVC, our method improves about 3.4% on ACC, 5.5% on NMI, 4.1% on ARI, 3.2% on F-score, 4.6% on Precision, 2.3% on Recall. Compared with DiMSC, our RDDMF method

Table 4 Clustering performance on Notting-Hill dataset

	ACC	NMI	ARI	F-score	Precision	Recall
BestSV	0.813±0.000	0.723±0.008	0.712±0.020	0.775±0.015	0.774±0.018	0.776±0.013
ConFea	0.673±0.033	0.628±0.028	0.612±0.041	0.696±0.032	0.699±0.032	0.693±0.031
ConPCA	0.733±0.008	0.632±0.009	0.598±0.015	0.685±0.012	0.691±0.010	0.680±0.014
Co-Reg	0.758±0.000	0.660±0.003	0.616±0.004	0.699±0.000	0.705±0.003	0.694±0.003
Co-Train	0.689±0.027	0.766±0.005	0.589±0.035	0.677±0.026	0.688±0.030	0.667±0.023
MultiNMF	0.831±0.001	0.752±0.001	0.762±0.000	0.815±0.000	0.804±0.001	0.824±0.001
NaMSC	0.752±0.013	0.730±0.002	0.666±0.004	0.738±0.005	0.746±0.002	0.730±0.011
DiMSC	0.843±0.021	0.799±0.001	0.787±0.001	0.834±0.001	0.822±0.005	0.836±0.009
DMVC	0.871±0.009	0.797±0.005	0.803±0.002	0.847±0.002	0.826±0.007	0.870±0.001
PLCMF	0.792±0.007	0.708±0.008	0.792±0.005	0.664±0.006	0.665±0.001	0.664±0.002
NMFCC	0.544±0.103	0.436±0.132	0.334±0.149	0.507±0.087	0.445±0.105	0.634±0.134
Ours	0.905±0.009	0.852±0.001	0.844±0.016	0.879±0.013	0.872±0.004	0.893±0.000

Table 5 Clustering performance on COIL20 dataset

	ACC	NMI	ARI	F-score	Precision	Recall
BestSV	0.811±0.010	0.904±0.003	0.794±0.007	0.805±0.008	0.768±0.015	0.846±0.003
ConFea	0.807±0.008	0.874±0.004	0.757±0.007	0.769±0.007	0.751±0.012	0.788±0.006
ConPCA	0.812±0.006	0.889±0.008	0.763±0.002	0.775±0.002	0.747±0.001	0.805±0.003
Co-Reg	0.631±0.016	0.887±0.003	0.639±0.008	0.663±0.006	0.496±0.005	0.845±0.005
Co-Train	0.689±0.008	0.771±0.009	0.472±0.011	0.597±0.009	0.503±0.011	0.747±0.019
MultiNMF	0.624±0.020	0.763±0.004	0.524±0.018	0.552±0.017	0.456±0.020	0.706±0.011
NaMSC	0.636±0.064	0.840±0.042	0.525±0.029	0.557±0.028	0.421±0.000	0.774±0.003
DiMSC	0.719±0.031	0.821±0.009	0.675±0.026	0.692±0.025	0.668±0.033	0.718±0.018
DMVC	0.809±0.001	0.926±0.001	0.776±0.014	0.784±0.009	0.713±0.027	0.885±0.011
PLCMF	0.676±0.004	0.781±0.004	0.696±0.005	0.636±0.064	0.596±0.007	0.675±0.003
NMFCC	0.789±0.066	0.891±0.033	0.742±0.069	0.756±0.065	0.701±0.085	0.824±0.036
Ours	0.820±0.005	0.931±0.000	0.808±0.001	0.816±0.004	0.759±0.001	0.887±0.000

Table 6 Ablation experiment results

methods	HandWritten		Extended yaleB		Notting-Hill		COIL20	
	ACC	NMI	ACC	NMI	ACC	NMI	ACC	NMI
RDDNMF,$\lambda = 0$	0.9595	0.9154	0.5918	0.5306	0.7033	0.6618	0.7755	0.9282
RDDNMF,$\beta = 0$	0.9590	0.9147	0.7111	0.6165	0.8891	0.8415	0.7750	0.9277
RDDNMF,$\mu = 0$	0.8115	0.8651	0.7424	0.6895	0.8424	0.8109	0.7736	0.9272
RDDNMF	0.9575	0.9214	0.7831	0.6969	0.8991	0.8415	0.8201	0.9314

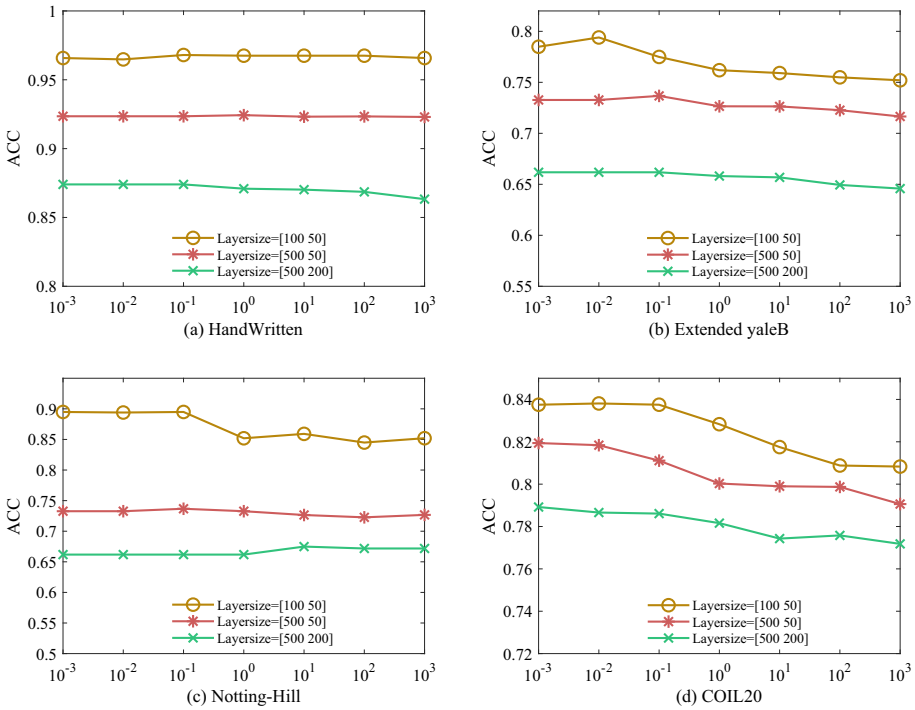

Fig. 2 ACC curves w.r.t parameter γ on four datasets with different layer size settings

improves around 6.2%, 5.3%,5.7%, 4.5%, 5.0%, 5.7% over DiMSC in terms of ACC, NMI, ARI, F-score, Precision and Recall, respectively. This is because the proposed RDDMF approach not only adopts the deep framework to learn the feature information of multi-view data, but also considers more prior knowledge than both DMVC and DiMSC. In addition, it should be noted that our approach is an unsupervised learning method without considering any label information among the multi-view data. Nevertheless, its performance is still better than the PLMCF approach with fully utilizing the supervised information such as labels.

Table 5 lists the performances of all multi-view clustering algorithms on the COIL20 dataset. In this dataset, we mainly analyze the performance influence of the model varied with the shooting angles. It can be seen from Table 5 that DMVC outperforms MultiNMF in terms of six metrics. The main reason is that MultiNMF is a shallow learning method that performs matrix decomposition only once. However, DMVC employs the deep matrix factorization method to effectively deal with the angle change of the images in multi-view clustering. In addition, it is clear to see that the performances of our proposed algorithm improve about 1.1% in ACC, 0.5% in NMI, 3.2% in ARI, 3.2% in F-score, 4.6% in Precision and 0.2% in Recall compared with DMVC. This is because our proposed RDDMF approach considers the manifold structure of both the feature and the data space, and imposes the sparse constraint on the coefficient matrix of each layer to cope with complex multi-view data, simultaneously.

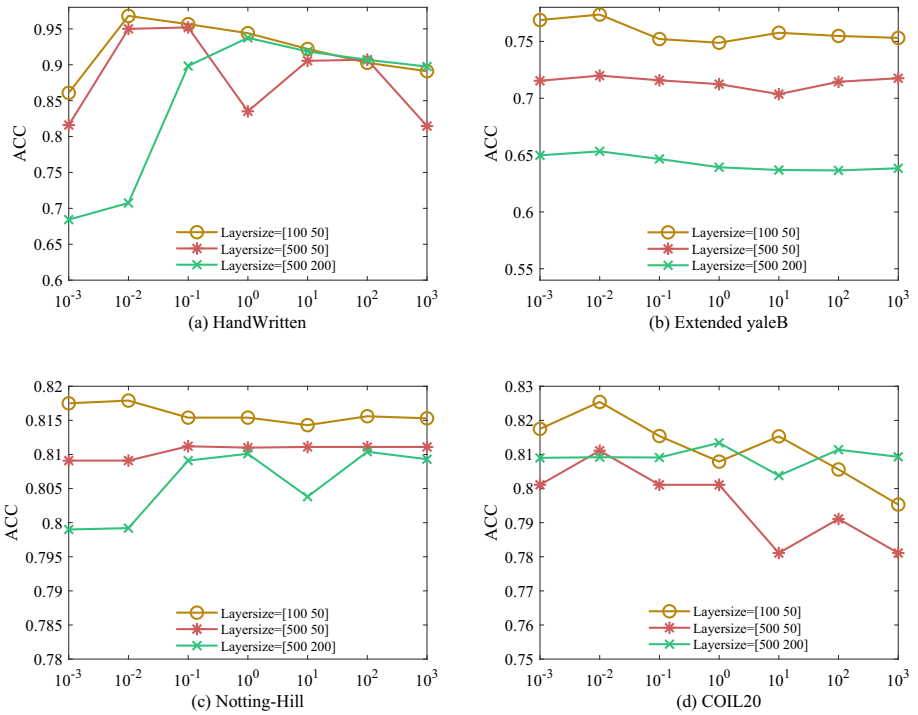

Fig. 3 ACC curves w.r.t parameter λ on four datasets with different layer size settings

4.5 Parametric Ablation Experiment

In the proposed RDDMF model, there are three regularization terms, namely feature graph regularizer, data graph regularizer and sparse regualarizer. The parameters β, λ and μ are used to control the contribution of the three regularizers in the model, respectively. Therefore, we have conducted ablation experiments to verify the effect of different regularization terms in the proposed model. Table 6 shows the ablation experimental results of the proposed method. From Table 6, we can see that the performances of the proposed model degrade in most cases by setting the parameter value of the regularization term to zero. Therefore, the results of the ablation experiments have manifested the effectiveness of our proposed RDDMF method.

4.6 Parameter Sensitivity Analysis

It can be seen from the objective function of the proposed method includes four important parameters γ, λ, β and μ. In addition, the experimental analysis of other common parameters, such as neighbor number k layer size l_r, were neglected due to space limitation. Figures 3, 4, 5 and 6 show the ACC results of our RDDMF approach concerning the parameters γ, λ, β and μ under three different layer size settings in the four datasets, respectively. The values of parameters γ, λ, β were set in the grid of 1×10^{-3}, 1×10^{-2}, 1×10^{-1}, 1, 1×10^{1}, 1×10^{2}, 1×10^{3} and the values of the parameter μ are set in the grid of 1×10^{-7}, 1×10^{-6}, 1×10^{-5}, 1×10^{-4}, 1×10^{-3}, 1×10^{-2}, 1×10^{-1}. Specifically, we varied the values of one parameter while fixing other parameters. It can be seen from Figs. 3, 4, 5 and 6 that

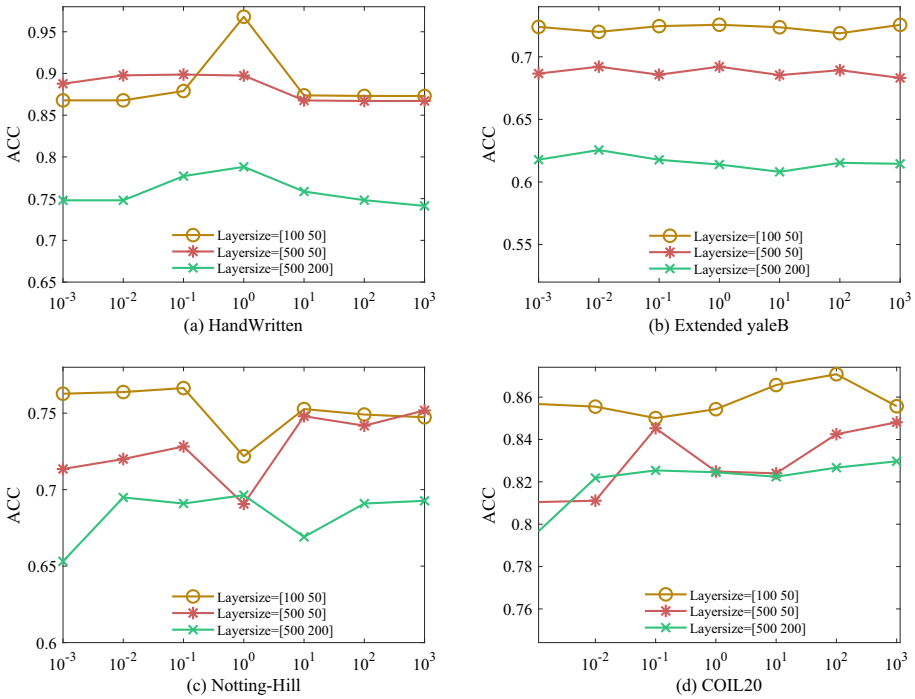

Fig. 4 ACC curves w.r.t parameter β on four datasets with different layer size settings

our proposed RDDMF approach can keep stable performance in a large range of parameter values.

4.7 Convergence Analysis

In this subsection, we carried out some experiments to further verify the convergence rate of the proposed RDDMF model. Figures 6, 7, 8 and 9 plot the convergence curves of RDDMF, NMFCC, PLCMF and DMVC on the HandWritten, Extended YaleB, Notting-Hill and COIL20 datasets. The x-axis denotes the values of the objective function and the y-axis represents the iteration times. It can be seen that RDDMF, PLCMF and DMVC can converge after within 10 iterations on four publicly available datasets. Therefore, it also shows the efficiency of our proposed optimization strategy.

5 Conclusion

In this paper, we propose a novel multi-view clustering approach, called RDDMF. Specifically, it effectively captures the hierarchical semantic information of multi-view data by adopting the deep matrix factorization. Besides, our RDDMF approach constructs two graph regularizers to model both the data manifold and the feature manifold hidden in multi-view data, and then integrates them into the matrix decomposition of each layer. In addition, to improve the robustness of the proposed approach, we impose the sparse constraint on the

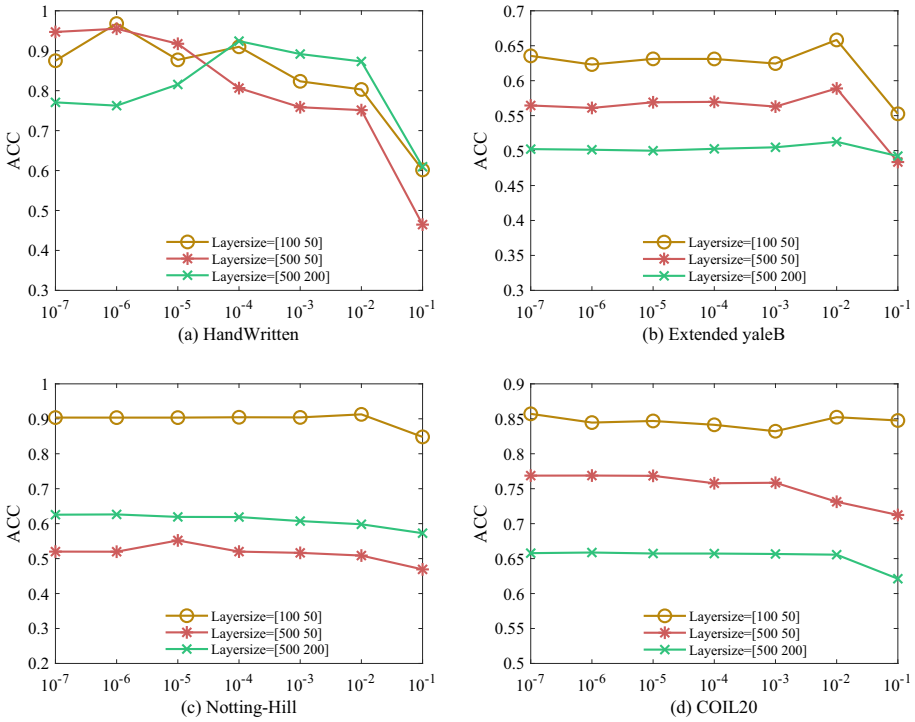

Fig. 5 ACC curves w.r.t parameter μ on four datasets with different layer size settings

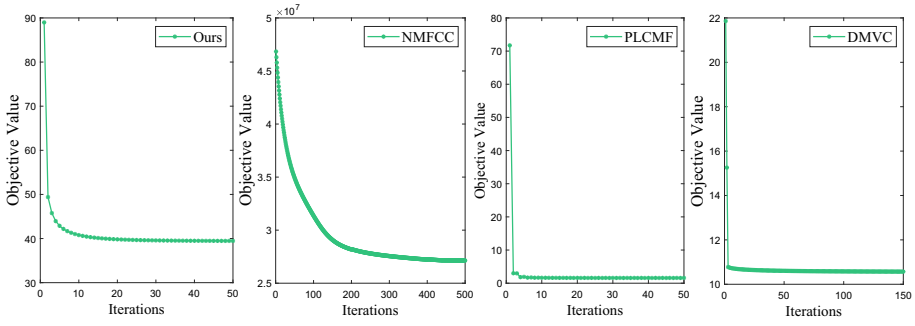

Fig. 6 The values of objective function w.r.t iteration times on HandWritten

coefficient matrix of each layer. Therefore, our RDDMF approach can fully utilize the prior knowledge embedded in multi-view data. Massive experimental results on four real datasets have shown that our RDDMF outperforms several state-of-the-art approaches in multi-view clustering.

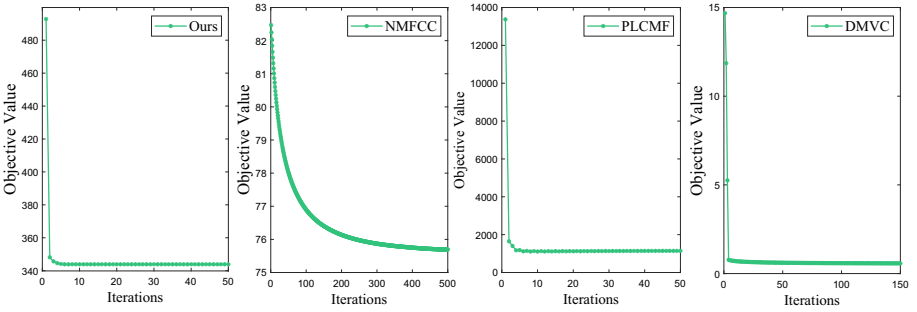

Fig. 7 The values of objective function w.r.t iteration times on Extended yaleB

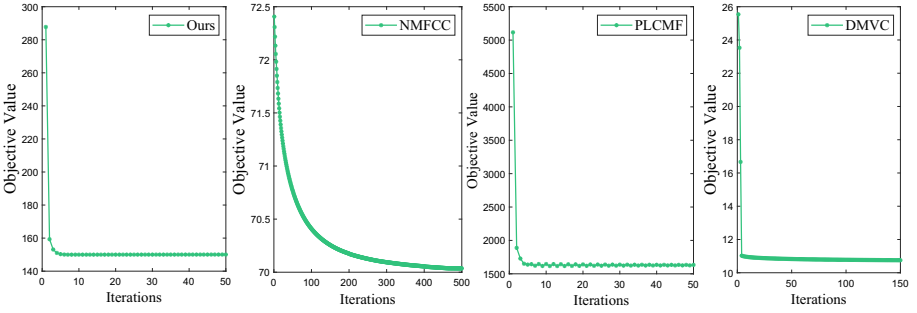

Fig. 8 The values of objective function w.r.t iteration times on Notting-Hill

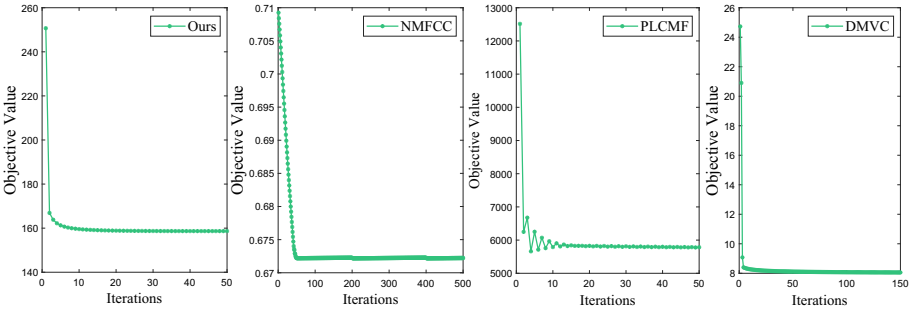

Fig. 9 The values of objective function w.r.t iteration times on COIL20

Acknowledgements This work was supported by the National Natural Science Foundation of China [Grant Nos. 61603159, 62162033, U21B2027, 62006097], Yunnan Provincial Major Science and Technology Special Plan Projects [Grant Nos. 202002AD080001, 202103AA080015], Yunnan Foundation Research Projects [Grant Nos. 202101AT070438, 202101BE070001-056], the Natural Science Foundation of Jiangsu Province (Grant No. BK20200593), Excellent Key Teachers of QingLan Project in Jiangsu Province.

References

1. Kumar A, Rai P, Daume H, (2011) Co-regularized multi-view spectral clustering. Adv Neural Inf Process Syst 24

2. Shu Z, Zuo F, Wu W, You C (2022) Dual local learning regularized NMF with sparse and orthogonal constraints. Appl Intell, pp. 1–15
3. Peng S, Ser W, Chen B, Lin Z (2021) Robust semi-supervised nonnegative matrix factorization for image clustering. Pattern Recogn 111:107683
4. Shu Z, Sun Y, Tang J, You C (2022) Adaptive graph regularized deep semi-nonnegative matrix factorization for data representation. Neural Process Lett pp. 1–19
5. Wang Q, He X, Jiang X, Li X (2020) Robust bi-stochastic graph regularized matrix factorization for data clustering. IEEE Trans Pattern Anal Mach Intell 44(1):390–403
6. Shu Z, Weng Z, Yu Z, You C, Liu Z, Tang S, Wu X (2022) Correntropy-based dual graph regularized nonnegative matrix factorization with lp smoothness for data representation. Appl Intell 52(7):7653–7669
7. Ye Q, Huang P, Zhang Z, Zheng Y, Fu L, Yang W (2021) Multiview learning with robust double-sided twin SVM. IEEE Trans Cybern
8. Fu L, Li Z, Ye Q, Yin H, Liu Q, Chen X, Fan X, Yang W, Yang G (2020) Learning robust discriminant subspace based on joint l2, p-and l2, s-norm distance metrics. IEEE Trans Neural Netw Learn Syst
9. Bai R, Huang R, Chen Y, Qin Y (2021) Deep multi-view document clustering with enhanced semantic embedding. Inf Sci 564:273–287
10. Liu P, Luo J, Chen X (2020) miRCom: Tensor completion integrating multi-view information to deduce the potential disease-related miRNA-miRNA pairs. IEEE/ACM Trans Comput Biol Bioinf
11. Jia X, Jing X-Y, Zhu X, Chen S, Du B, Cai Z, He Z, Yue D (2020) Semi-supervised multi-view deep discriminant representation learning. IEEE Trans Pattern Anal Mach Intell 43(7):2496–2509
12. Shi S, Nie F, Wang R, Li X (2021) Multi-view clustering via nonnegative and orthogonal graph reconstruction. IEEE Trans Neural Netw Learn Syst
13. Gao H, Nie F, Li X, Huang H (2015) Multi-view subspace clustering. In: Proceedings of the IEEE international conference on computer vision, pp. 4238–4246
14. Liu J, Wang C, Gao J, Han J (2013) Multi-view clustering via joint nonnegative matrix factorization. In: Proceedings of the 2013 SIAM international conference on data mining, pp. 252–260, SIAM
15. Chaudhuri K, Kakade SM, Livescu K, Sridharan K (2009) Multi-view clustering via canonical correlation analysis. In: Proceedings of the 26th annual international conference on machine learning, pp. 129–136
16. Wang D, Han S, Wang Q, He L, Tian Y, Gao X (2021) Pseudo-label guided collective matrix factorization for multiview clustering. IEEE Trans Cybern
17. Zhang Z, Liu L, Shen F, Shen HT, Shao L (2018) Binary multi-view clustering. IEEE Trans Pattern Anal Mach Intell 41(7):1774–1782
18. Li S-Y, Jiang Y, Zhou Z-H (2014) Partial multi-view clustering. In: Proceedings of the AAAI conference on artificial intelligence, p. 28
19. Zong L, Zhang X, Zhao L, Yu H, Zhao Q (2017) Multi-view clustering via multi-manifold regularized non-negative matrix factorization. Neural Netw 88:74–89
20. Trigeorgis G, Bousmalis K, Zafeiriou S, Schuller BW (2016) A deep matrix factorization method for learning attribute representations. IEEE Trans Pattern Anal Mach Intell 39(3):417–429
21. Zhao H, Ding Z, Fu Y (2017) Multi-view clustering via deep matrix factorization. In: Thirty-first AAAI conference on artificial intelligence
22. Huang S, Kang Z, Xu Z (2020) Auto-weighted multi-view clustering via deep matrix decomposition. Pattern Recogn 97:107015
23. Wei S, Wang J, Yu G, Domeniconi C, Zhang X (2020) Multi-view multiple clusterings using deep matrix factorization. Proc AAAI Conf Artif Intell 34:6348–6355
24. Lee D, Seung HS (2000) Algorithms for non-negative matrix factorization. Adv Neural Inf Process Syst, 13
25. Ding CH, Li T, Jordan MI (2008) Convex and semi-nonnegative matrix factorizations. IEEE Trans Pattern Anal Mach Intell 32(1):45–55
26. Meng Y, Shang R, Jiao L, Zhang W, Yang S (2018) Dual-graph regularized non-negative matrix factorization with sparse and orthogonal constraints. Eng Appl Artif Intell 69:24–35
27. Zhang Y-F, Xu C, Lu H, Huang Y-M (2009) Character identification in feature-length films using global face-name matching. IEEE Trans Multimed 11(7):1276–1288
28. Ng A, Jordan M, Weiss Y (2001) On spectral clustering: analysis and an algorithm. Adv Neural Inf Process Syst, 14
29. Hu H, Lin Z, Feng J, Zhou J (2014) Smooth representation clustering. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 3834–3841
30. Cao X, Zhang C, Fu H, Liu S, Zhang H (2015) Diversity-induced multi-view subspace clustering. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 586–594
31. Liang N, Yang Z, Li Z, Sun W, Xie S (2020) Multi-view clustering by non-negative matrix factorization with co-orthogonal constraints. Knowl-Based Syst 194:105582