Large-Scale Multi-View Clustering via Fast Essential Subspace Representation Learning

Qinghai Zheng

Abstract—**Large-scale Multi-View Clustering (LMVC) is a hot research problem in the fields of signal processing and machine learning, and many anchor-based multi-view subspace clustering algorithms are proposed in recent years. However, most existing methods usually concentrate on the issue of reducing the time cost and ignore the exploration of the complementary information during the clustering process. To this end, we propose a Fast Essential Subspace Representation Learning (FESRL) method for large-scale multi-view subspace clustering. Specifically, FESRL introduces the orthogonal transformation to investigate both the complementary and consensus information across multiple views. The essential subspace representation can be learned in a linear time cost. Experiments conducted on several benchmark datasets illustrate the competitiveness of the proposed method.**

Index Terms—**Large-scale multi-view subspace clustering, linear computational complexity.**

I. Introduction

With the technology development, multi-view data collected from various measurements or different domains are popular in real-world applications [1], [2], [3], [4], [5], [6], [7]. As an important and effective technique in the fields of signal processing and machine learning, multi-view subspace clustering has drawn widespread attention in recent years [8], [9], [10], [11]. Based on the self-expressiveness property, multi-view subspace clustering aims to achieve a promising subspace representation, which can be utilized to obtain clustering results based on spectral clustering algorithm [12], [13]. Many algorithms are proposed over the past few years [14], [15], [16], [17], [18], [19], [20], [21]. However, most existing multi-view subspace clustering methods suffer from the problem of high time cost. Actually, given a multi-view dataset with n samples, the time complexity of most existing methods is $\mathcal{O}(n^3)$, which greatly restricts the application prospect of multi-view subspace clustering, and it is necessary to develop technologies of large-scale multi-view subspace clustering.

Recently, several efforts have been made to obtain the multi-view subspace clustering results efficiently [22], [23], [24], [25]. For example, Large-scale Multi-View Subspace Clustering (LMVSC) introduces the anchor-based subspace representation learning framework [22], which learns multiple anchor-based subspace representations independently and achieves clustering results based on a landmark-based spectral clustering algorithm [26]. Inspired by LMVSC, Multi-view Structured Graph Learning (MSGL) [23] utilizes the rank constraint to guide the learning process of the shared anchor-based subspace representation. Regarding the time complexity, these methods can achieve the multi-view clustering results in a linear time complexity.

Although significant progress has been attained by recently proposed large-scale multi-view subspace clustering methods in real-world applications, most of them focus on the reduction of time cost while ignoring the investigation of complementary information among different views. For example, LMVSC independently learns subspace representations for multiple views and MSGL pursuits a common shared subspace representation of all views directly. Considering both the consensus and the complementary information, a more appropriate strategy is to assume that different anchor-based subspace representations have different forms while still leading to the same clustering results. For this purpose, this paper introduces a Fast Essential Subspace Representation Learning (FESRL) method for large-scale multi-view subspace clustering. Inspired by LMVSC and orthogonal invariance of Frobenius norm, FESRL introduces the orthogonal transformation, which guarantees the exploration of the complementary information, and learns the essential anchor-based subspace representation in a linear time cost. Comprehensive experiments conducted on several real-life multi-view datasets demonstrate the effectiveness of the proposed method.

The main contributions are summarized as follows:

- FESRL introduces the orthogonal transformation to learn the essential anchor-based subspace representation from multiple views in a linear time cost.
- FESRL explores both the consensus information and the complementary information of multi-view data during the anchor-based subspace representation learning process.
- We present an alternating direction iterative strategy for optimization and illustrate the superiority of the proposed method based on extensive experimental results.

II. Methodology

Given multi-view data $\{\boldsymbol{X}^{(i)} \in \mathbb{R}^{d_i \times n}\}_{i=1}^{v}$, with n samples and v views, we indicate the anchor matrix of the i-th view by $\boldsymbol{A}^{(i)} \in \mathbb{R}^{d_i \times m}$ and the anchor-based subspace representation of the i-th view by $\boldsymbol{Z}^{(i)} \in \mathbb{R}^{m \times n}$, in which m indicates the number of anchor samples, and d_i denotes the dimensionality of the i-th view.

A. Revisit of Large-Scale Multi-View Subspace Clustering

LMVSC [22] leverages the k-means to get anchor matrices $\{\boldsymbol{A}^{(i)}\}_{i=1}^{v}$ and achieves anchor-based subspace representations

Manuscript received 19 June 2022; revised 23 August 2022; accepted 23 August 2022. Date of publication 26 August 2022; date of current version 9 September 2022. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Minglun Gong.

The author is with the College of Computer and Data Science, Fuzhou University, Fuzhou 350108, China (e-mail: zhengqinghai@fzu.edu.cn).

Digital Object Identifier 10.1109/LSP.2022.3202108

independently:

$$\min_{\{\boldsymbol{Z}^{(i)}\}_{i=1}^{v}} \sum_{i=1}^{v} \left(\left\| \boldsymbol{X}^{(i)} - \boldsymbol{A}^{(i)} \boldsymbol{Z}^{(i)} \right\|_F^2 + \alpha \left\| \boldsymbol{Z}^{(i)} \right\|_F^2 \right),$$
$$\text{s.t. } \boldsymbol{Z}^{(i)} \geq \mathbf{0}, \boldsymbol{Z}^{(i)} \mathbf{1} = \mathbf{1}, \tag{1}$$

where $\|\cdot\|_F$ denotes the Frobenius norm [27]. Then it achieves multi-view clustering results by leveraging the landmark-based spectral clustering algorithm [26].

B. Fast Essential Subspace Representation Learning

Although LMVSC can achieve the promising clustering performance and has a linear time cost, it neglects the exploration of complementary information, which is important for multi-view clustering. To tackle this limitation, we proposed a Fast Essential Subspace Representation Learning (FESRL) method, which takes into account both issues of time cost and multi-view information investigation. To be specific, we introduce the following theorem according to the invariance of Frobenius norm firstly:

Theorem 1: Given matrices $\boldsymbol{P}$ and $\boldsymbol{U}$ with the proper sizes, if $\boldsymbol{U}$ is an orthogonal matrix, i.e., $\boldsymbol{U}^T\boldsymbol{U} = \boldsymbol{I}$, the following equation is satisfied:

$$\|\boldsymbol{P}\|_F^2 = \|\boldsymbol{U}\boldsymbol{P}\|_F^2 . \tag{2}$$

Based on Theorem 1, we can achieve the following orthogonal transformation:

$$\begin{aligned} &\min_{\boldsymbol{P}} \|\boldsymbol{P}\|_F^2 \\ &\Leftrightarrow \min_{\boldsymbol{P}} \|\boldsymbol{U}\boldsymbol{P}\|_F^2 , \text{ s.t. } \boldsymbol{U}^T\boldsymbol{U} = \boldsymbol{I} \\ &\Leftrightarrow \min_{\boldsymbol{V}} \|\boldsymbol{V}\|_F^2 , \text{ s.t. } \boldsymbol{U}^T\boldsymbol{U} = \boldsymbol{I}, \boldsymbol{V} = \boldsymbol{U}\boldsymbol{P}, \end{aligned} \tag{3}$$

Subsequently, the orthogonal transformation is imposed on the i-th view's anchor-based subspace representation $\boldsymbol{Z}^{(i)}$:

$$\begin{aligned} &\min_{\boldsymbol{Z}^{(i)}} \left\| \boldsymbol{Z}^{(i)} \right\|_F^2 \\ &\Leftrightarrow \min_{\boldsymbol{C}} \|\boldsymbol{C}\|_F^2 , \text{ s.t. } \boldsymbol{U}^{(i)^T} \boldsymbol{U}^{(i)} = \boldsymbol{I}, \boldsymbol{Z}^{(i)} = \boldsymbol{U}^{(i)} \boldsymbol{C}, \end{aligned} \tag{4}$$

where $\boldsymbol{C} \in \mathbb{R}^{m \times n}$ denotes the essential anchor-based subspace representation and $\boldsymbol{U}^{(i)} \in \mathbb{R}^{m \times m}$ is the orthogonal matrice for the orthogonal transformation of the i-th view.

Consequently, we construct the objective function of FESRL as follows:

$$\begin{aligned} &\min_{\{\boldsymbol{Z}^{\{i\}}, \boldsymbol{U}^{\{i\}}\}_{i=1}^{v}, \boldsymbol{C}} \sum_{i=1}^{v} \left\| \boldsymbol{X}^{(i)} - \boldsymbol{A}^{(i)} \boldsymbol{Z}^{(i)} \right\|_F^2 + \alpha \|\boldsymbol{C}\|_F^2 , \\ &\text{s.t. } \boldsymbol{Z}^{(i)} \geq \mathbf{0}, \boldsymbol{Z}^{(i)^T} \mathbf{1} = \mathbf{1}, \\ &\boldsymbol{Z}^{(i)} = \boldsymbol{U}^{(i)} \boldsymbol{C}, \boldsymbol{U}^{(i)^T} \boldsymbol{U}^{(i)} = \boldsymbol{I}, \boldsymbol{C} \geq \mathbf{0}, \boldsymbol{C}^T \mathbf{1} = \mathbf{1}. \end{aligned} \tag{5}$$

It can be observed that FESRL imposes the orthogonal transformation on the anchor-based subspace representations of all views. Unlike LMVSC, the anchor-based subspace representation learning process of one views can be associated with other views effectively, which can fully excavate the complementary information. Meanwhile, FESRL learns the essential anchor-based subspace representation $\boldsymbol{C}$ to investigate the underlying consensus information as well. Therefore, based on (5), both the consensus information and complementary information can be explored by our method. Once $\boldsymbol{C}$ is optimized, we can achieve clustering results by utilizing a landmark-based spectral clustering algorithm.

C. Optimizatin

In this section, we show the optimization process of (5). To be specific, the following augmented Lagrange function is constructed firstly:

$$\begin{aligned} &\mathcal{L}(\{\boldsymbol{Z}^{(i)}, \boldsymbol{U}^{(i)}, \boldsymbol{Y}^{(i)}\}_{i=1}^{v}, \boldsymbol{C}) \\ &= \sum_{i=1}^{v} \left\| \boldsymbol{X}^{(i)} - \boldsymbol{A}^{(i)} \boldsymbol{Z}^{(i)} \right\|_F^2 + \alpha \|\boldsymbol{C}\|_F^2 \\ &+ \sum_{i=1}^{v} \left(\left\langle \boldsymbol{Y}^{(i)}, \boldsymbol{Z}^{(i)} - \boldsymbol{U}^{(i)} \boldsymbol{C} \right\rangle + \frac{\mu}{2} \left\| \boldsymbol{Z}^{(i)} - \boldsymbol{U}^{(i)} \boldsymbol{C} \right\|_F^2 \right), \\ &\text{s.t. } \boldsymbol{Z}^{(i)} \geq \mathbf{0}, \boldsymbol{Z}^{(i)^T} \mathbf{1} = \mathbf{1}, \\ &\boldsymbol{U}^{(i)^T} \boldsymbol{U}^{(i)} = \boldsymbol{I}, \boldsymbol{C} \geq \mathbf{0}, \boldsymbol{C}^T \mathbf{1} = \mathbf{1}, \end{aligned} \tag{6}$$

where $\boldsymbol{Y}^{(i)}$ denotes the i-th view's Lagrange multiplier and μ is a penalty parameter with $\mu > 0$.

By leveraging an alternating direction iterative strategy, the following subproblems can be formulated and optimized:

1) Updating $\mathrm{Z}^{(i)}$*:* To update $\boldsymbol{Z}^{(i)}$, we fix other variables in (6) and construct the following subproblem:

$$\begin{aligned} &\min_{\boldsymbol{Z}^{(i)}} \left\| \boldsymbol{X}^{(i)} - \boldsymbol{A}^{(i)} \boldsymbol{Z}^{(i)} \right\|_F^2 \\ &+ \left\langle \boldsymbol{Y}^{(i)}, \boldsymbol{Z}^{(i)} - \boldsymbol{U}^{(i)} \boldsymbol{C} \right\rangle + \frac{\mu}{2} \left\| \boldsymbol{Z}^{(i)} - \boldsymbol{U}^{(i)} \boldsymbol{C} \right\|_F^2 , \\ &\text{s.t. } \boldsymbol{Z}^{(i)} \geq \mathbf{0}, \boldsymbol{Z}^{(i)^T} \mathbf{1} = \mathbf{1}, \end{aligned} \tag{7}$$

Considering the j-th column of $\boldsymbol{Z}^{(i)}$, i.e., $\boldsymbol{z}_{:,j}^{(i)}$, we rewrite (7) and get the following quadratic programming [28]:

$$\min_{\boldsymbol{z}_{:,j}^{(i)}} \frac{1}{2} \boldsymbol{z}_{:,j}^{(i)^T} \boldsymbol{H}_z \boldsymbol{z}_{:,j}^{(i)} + \boldsymbol{f}_z \boldsymbol{z}_{:,j}^{(i)}, \text{ s.t. } \boldsymbol{z}_{:,j}^{(i)} \geq \mathbf{0}, \boldsymbol{z}_{:,j}^{(i)^T} \mathbf{1} = 1, \tag{8}$$

where $\boldsymbol{H}_z = 2\boldsymbol{A}^T\boldsymbol{A} + \boldsymbol{I}$ and $\boldsymbol{f}_z$ is a vector defined as follows:

$$\boldsymbol{f}_z = \boldsymbol{y}_{:,j}^{(i)^T} - 2\boldsymbol{x}_{:,j}^{(i)^T} \boldsymbol{A} - \alpha \boldsymbol{c}_{:,j}^{(i)^T} \boldsymbol{U}^{(i)^T} \tag{9}$$

where $\boldsymbol{y}_{:,j}$, $\boldsymbol{x}_{:,j}$, and $\boldsymbol{c}_{:,j}$ indicate the j-th columns of $\boldsymbol{Y}$, $\boldsymbol{X}$, and $\boldsymbol{C}$. Eq (8) can be optimized effectively by employing the *quadprog* function provided in Matlab.[1]

2) Updating $\mathrm{U}^{(i)}$*:* By fixing other variables in (6), the subproblem of updating $\boldsymbol{U}^{(i)}$ can be written as follows:

$$\min_{\boldsymbol{U}^{(i)}} \left\| \left(\boldsymbol{Z}^{(i)} + \frac{\boldsymbol{Y}^{(i)}}{\mu} \right) - \boldsymbol{C}^T \boldsymbol{U}^{(i)^T} \right\|_F^2 , \text{ s.t. } \boldsymbol{U}^{(i)^T} \boldsymbol{U}^{(i)} = \boldsymbol{I}, \tag{10}$$

which is a typical orthogonal Procrustes problem and can be optimized with a closed-form solution according to [29].

[1] [Online]. Available: https://www.mathworks.com/help/optim/ug/quadprog.html

TABLE I
STATISTICAL INFORMATION OF DATASETS

Dataset	# samples	# clusters	# dimensionalities
UCI	2000	10	240, 76, 6
NH-Face	4660	5	6750, 2000, 3304
Caltech-101	9144	102	48,40, 254, 512, 928
SUNRGBD	10335	45	4096, 4096
NUS-WIDE-OBJ	30000	31	65, 226, 145, 74, 129

Algorithm 1: Optimization algorithm for FESRL.

Input: Multi-view data: $\{\boldsymbol{X}^{(v)}\}_{v=1}^{V}$, tradeoff parameter: α, numer of anchors: m.
Initialization: Initialize $\{\boldsymbol{Z}^{(i)}, \boldsymbol{U}^{(i)}, \boldsymbol{Y}^{(i)}\}_{i=1}^{v}$ and $\boldsymbol{C}$.
Optimization:
Achieve anchor matrices $\{\boldsymbol{A}^{(i)}\}_{i=1}^{v}$ via k-means,
while *not converge* **do**
  for $i = 1 : v$ **do**
    Update $\boldsymbol{Z}^{(i)}$ and $\boldsymbol{U}^{(i)}$ with Eq. (8) and (10),
  end
  Update $\boldsymbol{C}$ with Eq. (11),
  for $i = 1 : v$ **do**
    Update $\boldsymbol{Y}^{(i)}$ with Eq. (13),
  end
  Update μ with Eq. (13),
end
Output: Essential subspace representation: $\boldsymbol{C}$.

3) Updating C*:* Similar to the subproblem of updating $\boldsymbol{Z}^{(i)}$, we construct the following subproblem:

$$\min_{\boldsymbol{c}_{:,j}} \frac{1}{2}\boldsymbol{c}_{:,j}^{T}\boldsymbol{H}_{c}\boldsymbol{c}_{:,j} + \boldsymbol{f}_{c}\boldsymbol{c}_{:,j}, \text{ s.t. } \boldsymbol{c}_{:,j} \geq \boldsymbol{0}, \boldsymbol{c}_{:,j}^{T}\boldsymbol{1} = 1. \quad (11)$$

where $\boldsymbol{H}_c = (2\alpha + v\mu)\boldsymbol{I}$, and

$$\boldsymbol{f}_c = -\sum_{i=1}^{v} \boldsymbol{y}_{:,j}^{(i)^T}\boldsymbol{U}^{(i)} + \mu\boldsymbol{z}_{:,j}^{(i)^T}\boldsymbol{U}^{(i)}. \quad (12)$$

We employ the *quadprog* function for optimization as well.

4) Updating $\mathrm{Y}^{(i)}$*:* We update $\boldsymbol{Y}^{(i)}$ and α as follows:

$$\begin{aligned}\boldsymbol{Y}^{(i)} &= \boldsymbol{Y}^{(i)} + \mu(\boldsymbol{Z}^{(i)} - \boldsymbol{U}^{(i)}\boldsymbol{C}),\\ \mu &= \min(\rho\mu, \mu_{\max}),\end{aligned} \quad (13)$$

where ρ is a fixed scale and we set $\rho = 2$ in this paper, $\mu_{\max}$ denotes the maximum value of μ [30].

D. Time Complexity and Convergence

Given a multi-view dataset with n samples, v views, and k clusters, we indicate the number of optimization iterations by p, and utilize q to denote the number of k-means iterations for anchor selection. The main time complexity consists of three parts: 1) Anchor matrices construction, which requires $\mathcal{O}(vnmq)$; 2) Optimizations of $\{\boldsymbol{Z}^{(i)}, \boldsymbol{U}^{(i)}, \boldsymbol{Y}^{(i)}\}_{i=1}^{v}$, μ and $\boldsymbol{C}$, which needs $\mathcal{O}(2vnm^3)$ for $\{\boldsymbol{Z}^{(i)}\}_{i=1}^{v}$, needs $\mathcal{O}(vm^3)$ for $\{\boldsymbol{U}^{(i)}\}_{i=1}^{v}$, needs $\mathcal{O}(vnm^3)$ for $\boldsymbol{C}$, needs $\mathcal{O}(vnm)$ for $\{\boldsymbol{Y}^{(i)}\}_{i=1}^{v}$, and needs $\mathcal{O}(n)$ for μ; 3) Landmark-based spectral clustering on $\boldsymbol{C}$, which requires $\mathcal{O}(m^3 + nm^2)$. Since v, m, p, k and q are all far less than n, we can conclude that the time complexity of FESRL is $\mathcal{O}(n)$.

Regarding the convergence property of our method, the theoretical proof is difficult, since more than two subproblems are involved during the optimization. Fortunately, experimental results in the next section show that the proposed FESRL has the promising convergence property.

III. EXPERIMENTS

To evaluate the proposed FESRL, comprehensive experiments are conducted in this section. All experiments are performed on MATLAB R2018a, the experimental environment is a personal computer with the 11-th Gen 2.70 GHz Intel Core i5-11400H and 16 GB RAM.

A. Datasets, Baselines, and Evaluation Metrics

The following five benchmark multi-view datasets are used in this section, including UCI[2], NH-Face [31], Caltech-101[3], SUNRGBD[4], and NUS-WIDE-OBJ[5] (as shown in Table I). As for the baselines, two single-view methods, namely Large-scale Subspace Clustering (LSuC) [22] and Large-scale Spectral Clustering (LSpC) [26], and three large-scale multi-view clustering methods, including BMVC [32], LMVSC [22], and OPMC [33], are used here for comparison. For LSpC and LSuC, the subscript BSV indicates that the algorithm is conducted on each view and we provide the best clustering results, the subscript FC indicates that the algorithm is performed on the concatenated feature to achieve clustering results. For all comparison methods, we employ the settings recommended in their works. For our method, we set $m = 3k$, where k is the number of clusters, and we select α from $\{0.01, 0.1, \ldots, 1000\}$.

Regarding the evaluation metrics, Normalized Mutual Information (NMI), Purity, and ACCuracy (ACC) are employed to quantitatively describe the clustering performance.

B. Experimental Results

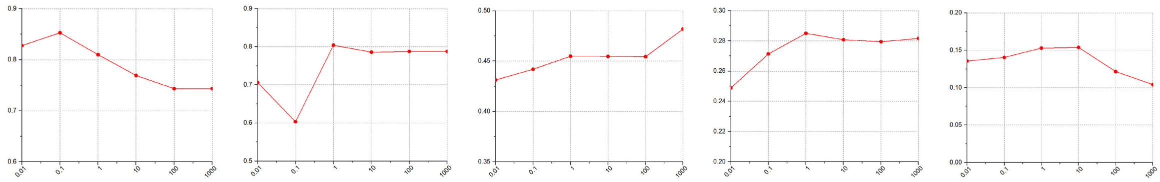
We present experimental results on real-world datasets here. Quantitatively, Table II shows the clustering results in metrics of NMI, purity, and ACC. In general, the promising clustering performance can be achieved by our FESRL in all metrics. According to the results in Table II, we get the following observations: 1) Compared with large-scale single-view clustering algorithms, namely LSuC and LSpC, the proposed method can achieve remarkable improvement in most cases. Interestingly, $\mathrm{LSuC}_{\mathrm{BSV}}$ and $\mathrm{LSpC}_{\mathrm{BSV}}$ achieve better clustering results than $\mathrm{LSuC}_{\mathrm{FC}}$ and $\mathrm{LSpC}_{\mathrm{FC}}$ in more than half cases, which means a suitable way to excavate multi-view information is significant to get promising clustering results. Meanwhile, it also verifies the effectiveness of the proposed method. 2) Compared with other large-scale multi-view clustering algorithms, the competitive clustering performance can be attained by our method as well. Furthermore, the clustering results of the proposed method can

[2][Online]. Available: https://archive.ics.uci.edu/ml/datasets/optical+recognition+of+handwritten+digits

[3][Online]. Available: http://www.vision.caltech.edu/Image_Datasets/Caltech101

[4][Online]. Available: http://rgbd.cs.princeton.edu/

[5][Online]. Available: https://lms.comp.nus.edu.sg/wp-content/uploads/2019/research/nuswide/NUS-WIDE.html



(a) UCI (b) NF-Face (c) Caltech101 (d) SUNRGBD (e) NUS-WIDE-OBJ

Fig. 1. The experimental results in the metric of NMI with respect to different α.

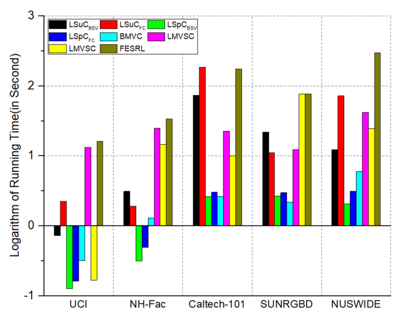
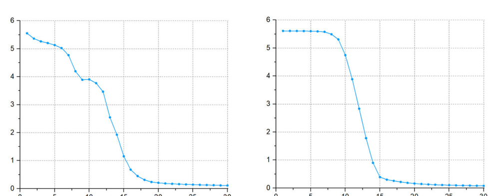


Fig. 2. The time-consuming of different methods.



(a) UCI (b) NF-Face

Fig. 3. The convergence curves on (a) UCI and (b) NH-Face.

be better than the clustering results of LMVSC in all cases, the underlying reason is that our FESRL introduces the orthogonal transformation and explores both the consensus and complementary information effectively.

We also show the running time of these methods in Fig. 2. Although other large-scale methods have slightly less running time, the clustering results of FESRL are significantly better than other methods in most cases. Therefore, considering both the clustering results and running time, our method is a good choice for large-scale multi-view subspace clustering.

C. Parameter Sensitivity and Convergence

In FESRL, m is fixed to $m = 3k$, where k denotes the number of clusters, and α is selected from $\{0.01, 0.1, \ldots, 1000\}$. To analyze the parameter sensitivity w.r.t. α, we provide the clustering results with different values of α in Fig. 1. It can be observed that FESRL is robust to α within a wide range.

Regarding the convergence, we take results on UCI and NH-Face for examples, and use the following loss:

$$\mathcal{L}_{loss} = \sum_{i=1}^{v} \left\| \boldsymbol{Z}^{(i)} - \boldsymbol{U}^{(i)}\boldsymbol{C} \right\|_{\infty}. \tag{14}$$

TABLE II

EXPERIMENTAL RESULTS ON FIVE BENCHMARK MULTI-VIEW DATASETS. WE HIGHLIGHT AND UNDERLINE THE BEST AND THE SECOND-BEST RESULTS RESPECTIVELY

Dataset	Method	NMI	purity	ACC
UCI	LSuC_{BSV}	0.6996	0.7121	0.6405
	LSuC_{FC}	0.6411	0.7250	0.6985
	LSpC_{BSV}	0.6267	0.6685	0.6515
	LSpC_{FC}	0.4590	0.4432	0.4171
	BMVC	0.7960	0.8150	0.7825
	LMVSC	0.7352	0.7670	0.7241
	OPMC	0.7882	0.8345	0.8345
	FESRL	**0.8530**	**0.9235**	**0.9235**
NH-Face	LSuC_{BSV}	0.6723	0.7994	0.7163
	LSuC FC	0.5947	0.7142	0.6137
	LSpC_{BSV}	0.6532	0.7888	0.6442
	LSpC_{FC}	0.5533	0.7139	0.662
	BMVC	0.0706	0.3856	0.3191
	LMVSC	0.6814	0.8056	0.7255
	OPMC	0.7656	0.8318	0.7034
	FESRL	**0.8041**	**0.8573**	**0.8498**
Caltech-101	LSuC_{BSV}	0.3915	0.3602	0.1836
	LSuC_{FC}	0.2920	0.2624	0.1273
	LSpC_{BSV}	0.4398	0.4161	0.2081
	LSpC_{FC}	0.3133	0.2879	0.1229
	BMVC	0.4426	0.414	0.2318
	LMVSC	0.4188	0.3952	0.2016
	OPMC	0.4658	0.4507	0.2652
	FESRL	**0.4819**	**0.4616**	**0.2734**
SUNRGBD	LSuC_{BSV}	0.254	0.3789	0.1711
	LSuC_{FC}	0.2825	0.3975	0.1797
	LSpC_{BSV}	0.2531	0.3754	0.1760
	LSpC_{FC}	0.2741	0.3822	0.1911
	BMVC	0.2314	0.3609	0.1612
	LMVSC	0.2607	0.3801	0.1849
	OPMC	0.2567	0.3988	0.1869
	FESRL	**0.2850**	**0.4074**	**0.2036**
NUS-WIDE-OBJ	LSuC_{BSV}	0.1178	0.2267	0.1428
	LSuC_{FC}	0.1388	0.2388	0.1495
	LSpC_{BSV}	0.1140	0.2346	0.1147
	LSpC_{FC}	0.1316	0.2434	0.1347
	BMVC	**0.1573**	**0.2729**	0.1478
	LMVSC	0.1256	0.2413	0.1348
	OPMC	0.1472	0.2635	0.1534
	FESRL	0.1539	0.2592	**0.1594**

As shown in Fig. 3, FESRL can quickly converge within 20 iterations, which empirically verifies the convergence property of the proposed method.

IV. CONCLUSION

This paper proposes a Fast Essential Subspace Representation Learning (FESRL) method, which fully excavates and exploits the important complementary information for large-scale multi-view subspace clustering. By introducing the orthogonal transformations on anchor-based subspace representations of different views, the proposed FESRL can learn the essential subspace representation in a linear time cost and fully explores the underlying multi-view information. Compared with several state-of-the-arts, promising clustering results can be achieved by the proposed method.

REFERENCES

[1] C. Xu, D. Tao, and C. Xu, “A survey on multi-view learning,” 2013, *arXiv:1304.5634*.

[2] J. Zhao, X. Xie, X. Xu, and S. Sun, “Multi-view learning overview: Recent progress and new challenges,” *Inf. Fusion*, vol. 38, pp. 43–54, 2017.

[3] Q. Yin et al., “Multi-view label prediction for unsupervised learning person re-identification,” *IEEE Signal Process. Lett.*, vol. 28, pp. 1390–1394, 2021.

[4] W. Hao, S. Pang, J. Zhu, and Y. Li, “Self-weighting and hypergraph regularization for multi-view spectral clustering,” *IEEE Signal Process. Lett.*, vol. 27, pp. 1325–1329, 2020.

[5] J. Ye, Q. Xiao, J. Wang, H. Zhang, J. Deng, and Y. Lin, “Cosleep: A multi-view representation learning framework for self-supervised learning of sleep stage classification,” *IEEE Signal Process. Lett.*, vol. 29, pp. 189–193, 2021.

[6] C. Tang, Z. Li, J. Wang, X. Liu, W. Zhang, and E. Zhu, “Unified one-step multi-view spectral clustering,” *IEEE Trans. Knowl. Data Eng.*, early access, doi: 10.1109/TKDE.2022.3172687.

[7] X. Yang, C. Deng, F. Zheng, J. Yan, and W. Liu, “Deep spectral clustering using dual autoencoder network,” in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 4066–4075.

[8] G. Chao, S. Sun, and J. Bi, “A survey on multiview clustering,” *IEEE Trans. Artif. Intell.*, vol. 2, no. 2, pp. 146–168, Apr. 2021, doi: 10.1109/TAI.2021.3065894.

[9] Z. Xue, J. Du, D. Du, G. Li, Q. Huang, and S. Lyu, “Deep constrained low-rank subspace learning for multi-view semi-supervised classification,” *IEEE Signal Process. Lett.*, vol. 26, no. 8, pp. 1177–1181, Aug. 2019.

[10] Q. Zheng, J. Zhu, H. Tang, X. Liu, Z. Li, and H. Lu, “Generalized label enhancement with sample correlations,” *IEEE Trans. Knowl. Data Eng.*, early access, doi: 10.1109/TKDE.2021.3073157.

[11] X. Yang, C. Deng, T. Liu, and D. Tao, “Heterogeneous graph attention network for unsupervised multiple-target domain adaptation,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 4, pp. 1992–2003, Apr. 2022.

[12] E. Elhamifar and R. Vidal, “Sparse subspace clustering: Algorithm, theory, and applications,” *IEEE Trans. Pattern Anal. Mach. Intell.,* vol. 35, no. 11, pp. 2765–2781, Nov. 2013.

[13] C. Tang et al., “Learning a joint affinity graph for multiview subspace clustering,” *IEEE Trans. Multimedia*, vol. 21, no. 7, pp. 1724–1736, Jul. 2019.

[14] Q. Zheng, J. Zhu, Z. Li, S. Pang, J. Wang, and Y. Li, “Feature concatenation multi-view subspace clustering,” *Neurocomputing*, vol. 379, pp. 89–102, 2020.

[15] L. Zhao, J. Zhang, Q. Wang, and Z. Chen, “Dual alignment self-supervised incomplete multi-view subspace clustering network,” *IEEE Signal Process. Lett.*, vol. 28, pp. 2122–2126, 2021.

[16] Y. Xie, D. Tao, W. Zhang, Y. Liu, L. Zhang, and Y. Qu, “On unifying multi-view self-representations for clustering by tensor multi-rank minimization,” *Int. J. Comput. Vis.*, vol. 126, no. 11, pp. 1157–1179, 2018.

[17] C. Zhang et al., “Generalized latent multi-view subspace clustering,” *IEEE Trans. Pattern Anal. Mach. Intell.,* vol. 42, no. 1, pp. 86–99, Jan. 2020.

[18] Z. Kang et al., “Partition level multiview subspace clustering,” *Neural Netw.*, vol. 122, pp. 279–288, 2020.

[19] Q. Zheng, J. Zhu, Y. Ma, Z. Li, and Z. Tian, “Multi-view subspace clustering networks with local and global graph information,” *Neurocomputing*, vol. 449, pp. 15–23, 2021.

[20] Z. Li, D. Wu, F. Nie, R. Wang, Z. Sun, and X. Li, “Multi-view clustering based on invisible weights,” *IEEE Signal Process. Lett.*, vol. 28, pp. 1051–1055, 2021.

[21] X. Yang, C. Deng, Z. Dang, and D. Tao, “Deep multiview collaborative clustering,” *IEEE Trans. Neural Netw. Learn. Syst.*, early access, doi: 10.1109/TNNLS.2021.3097748.

[22] Z. Kang, W. Zhou, Z. Zhao, J. Shao, M. Han, and Z. Xu, “Large-scale multi-view subspace clustering in linear time,” in *Proc. AAAI Conf. Artif. Intell.*, 2020, vol. 34, no. 04, pp. 4412–4419.

[23] Z. Kang, Z. Lin, X. Zhu, and W. Xu, “Structured graph learning for scalable subspace clustering: From single view to multiview,” *IEEE Trans. Cybern.*, vol. 52, no. 9, pp. 8976–8986, Sep. 2021.

[24] J. Fan, “Large-scale subspace clustering via k-factorization,” in *Proc. 27th ACM SIGKDD Conf. Knowl. Discov. Data Mining*, 2021, pp. 342–352.

[25] M. Sun et al., “Scalable multi-view subspace clustering with unified anchors,” in *Proc. 29th ACM Int. Conf. Multimedia*, 2021, pp. 3528–3536.

[26] D. Cai and X. Chen, “Large scale spectral clustering via landmark-based sparse representation,” *IEEE Trans. Cybern.*, vol. 45, no. 8, pp. 1669–1680, Aug. 2015.

[27] P. Ji, M. Salzmann, and H. Li, “Efficient dense subspace clustering,” in *Proc. IEEE Winter Conf. Appl. Comput. Vis.*, 2014, pp. 461–468.

[28] N. Gould and P. L. Toint, “Preprocessing for quadratic programming,” *Math. Program.*, vol. 100, no. 1, pp. 95–132, 2004.

[29] P. H. Schönemann, “A generalized solution of the orthogonal procrustes problem,” *Psychometrika*, vol. 31, no. 1, pp. 1–10, 1966.

[30] X. Ren and Z. Lin, “Linearized alternating direction method with adaptive penalty and warm starts for fast solving transform invariant low-rank textures,” *Int. J. Comput. Vis.*, vol. 104, no. 1, pp. 1–14, 2013.

[31] Y.-F. Zhang, C. Xu, H. Lu, and Y.-M. Huang, “Character identification in feature-length films using global face-name matching,” *IEEE Trans. Multimedia*, vol. 11, no. 7, pp. 1276–1288, Nov. 2009.

[32] Z. Zhang, L. Liu, F. Shen, H. T. Shen, and L. Shao, “Binary multi-view clustering,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 7, pp. 1774–1782, Jul. 2019.

[33] J. Liu et al., “One-pass multi-view clustering for large-scale data,” in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2021, pp. 12344–12353.