An Efficient Newton-Based Method for Sparse Generalized Canonical Correlation Analysis

Xinrong Li, Xianchao Xiu, *Member, IEEE*, Wanquan Liu, *Senior Member, IEEE*, and Zhonghua Miao

***Abstract*—Generalized canonical correlation analysis (GCCA) that aims to deal with multi-view data has attracted extensive attention in signal processing. To improve the representation performance, this letter proposes a new sparsity constrained GCCA (SCGCCA). Technically, it integrates the $\ell_{2,0}$-norm constrained optimization into GCCA, which has not been investigated in the literature. Compared with the existing $\ell_{2,1}$-norm regularized GCCA, the proposed SCGCCA can not only exploit the similarity information belonging to the same features but also determine the number of extracted features. Although it is a nonconvex minimization problem, an efficient alternating minimization algorithm can be designed. Furthermore, a Newton hard thresholding pursuit technique is developed to accelerate the convergence tremendously. Empirical studies suggest both the effectiveness and efficiency of the proposed SCGCCA comparing with the existing GCCA and its variants. In particular, the speed can be increased by 150 times for the simulated dataset.**

***Index Terms*—Multi-view learning, generalized canonical correlation analysis (GCCA), sparse optimization, Newton method.**

I. Introduction

With the development of modern sensor technology, large-scale data appear to be multi-view [1]. Since different views provide coherent but complementary prior knowledge, making full use of multi-view data can improve the representation performance. According to [2], multi-view learning methods can be mainly classified into three types: co-training [3], multiple kernel learning [4], and subspace representation learning [5], [6]. An effective multi-view learning method is referred to as generalized canonical correlation analysis (GCCA) which learns a shared space by aggregating the variables from each view [7]. Now, it has been widely used in signal processing [8], [9], biomedical engineering [10], [11], and information sciences [12]–[15].

During the last few decades, many variants of GCCA have been proposed, ranging from graph GCCA [16], discriminative GCCA [17], structured GCCA [18], nonlinear GCCA [19], to deep GCCA [20]. However, in high-dimensional settings, all the aforementioned GCCA may derive a solution in lack of physical interpretability because not all the canonical variables are identically informative for representing the data. To overcome this problem, the $\ell_{2,1}$-norm (sum of ℓ_2-norm of rows) is imposed to the GCCA objective function [21]. It is proved that sparse GCCA (SGCCA) can improve the interpretability by removing some unimportant features [22]. Further, an elastic net regularized version of SGCCA, called EGCCA, was proposed in [23].

However, two issues remain open. On one hand, although fast solvers are proposed, such as the two-stage method [24] and the linearized Bregman method [25], soft-thresholding operators for all the elements are always involved, which brings huge computational load and thus limits the scalability to large-scale applications. On the other hand, the regularization parameters should be tuned carefully, which is not practical for real applications because there often exists physical demands. For example, in fault diagnosis, only s variables need to be extracted [26]. As another example, in surveillance videos, the background matrix is low-rank [27]. Therefore, a fast and efficient variant of SGCCA needs to be considered.

In sparse learning, the ℓ_0-norm (number of nonzero entries) constrained optimization has shown its advantages in feature selection since it is the original description of sparsity. In algorithms, a greedy method, called iterative hard thresholding, can be applied to seek approximate solutions [28]. To improve the convergence rate, a gradient hard thresholding pursuit algorithm was designed [29]. Very recently, a Newton hard thresholding pursuit (NHTP) algorithm was proposed and proved to have global and quadratic convergence guarantee theoretically [30], and the promising performance was demonstrated for sparse logistic regression [31]. It is admitted that compared with convex or nonconvex relaxations, the ℓ_0-norm constrained optimization has faster optimization algorithms and easier variable selection. However, the integration with GCCA has not been investigated yet. Considering the necessity of joint sparsity, a natural question is whether the ℓ_0-norm can be extended to $\ell_{2,0}$-norm, and then incorporated with GCCA.

Inspired by the above discussion, this letter proposes a new sparse constrained GCCA (SCGCCA) for learning multi-view representations, which, to the best of the authors' knowledge, is the first work to integrate the $\ell_{2,0}$-norm into such a GCCA framework. The main contributions of the current work are:

1) A novel multi-view learning model is constructed by introducing the $\ell_{2,0}$-norm constrained optimization.
2) A Newton hard thresholding pursuit-based algorithm is developed to solve the proposed minimization problem.
3) A number of experiments are conducted to show the superiority of the proposed model and algorithm.

Manuscript received September 29, 2021; revised November 6, 2021; accepted November 10, 2021. Date of publication November 18, 2021; date of current version January 24, 2022. This work was supported in part by the National Natural Science Foundation of China under Grant 12001019 and in part by the Natural Science Foundation of Shanghai under Grant 19ZR1420800. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Yipeng Liu. *(Corresponding author: Xianchao Xiu.)*

Xinrong Li is with the School of Computer and Information Technology, Beijing Jiaotong University, Beijing 100044, China (e-mail: lixinrong0827@163.com).

Xianchao Xiu and Zhonghua Miao are with the School of Mechatronic Engineering and Automation, Shanghai University, Shanghai 200444, China (e-mail: xcxiu@shu.edu.cn; zhhmiao@shu.edu.cn).

Wanquan Liu is with the School of Intelligent Systems Engineering, Sun Yat-sen University, Guangzhou 510275, China (e-mail: liuwq63@mail.sysu.edu.cn).

Digital Object Identifier 10.1109/LSP.2021.3129170

II. Sparsity Constrained GCCA

Given multi-view data $\mathbf{X}_v \in \mathbb{R}^{n\times d_v} (v = 1, \ldots, M)$, the proposed SCGCCA seeks a shared representation $\mathbf{U} \in \mathbb{R}^{n\times d}$ and projections $\mathbf{P}_v \in \mathbb{R}^{d_v \times d}$ with maximal correlation and sparse prior. Mathematically, it is defined as

$$\min_{\mathbf{U},\mathbf{P}_v} \sum_{v=1}^{M} \|\mathbf{U} - \mathbf{X}_v\mathbf{P}_v\|_F^2$$
$$\text{s.t. } \mathbf{U}^T\mathbf{U} = \mathbf{I}_d,\ \|\mathbf{P}_v\|_{2,0} \leq s_v, \tag{1}$$

where $\|\mathbf{P}_v\|_{2,0} := |\{i : \|(\mathbf{P}_v)_{i,:}\|_2 \neq 0\}|$ counts the number of nonzero rows, of which $(\cdot)_{i,:}$ denotes the i-th row and $|\cdot|$ denotes the cardinality, and $s_v \ll n$ is a positive integer, which can be chosen by physical laws or cross-validation techniques.

From the aspect of optimization, problem (1) is not easy to solve because all the constraints are nonconvex and the objective function involves more than two variables. According to the alternating minimization technique, it can be computed by solving one variable with the others fixed [32]. Therefore, the detailed procedure can be provided in Algorithm 1.

Algorithm 1: Optimization Algorithm for Solving (1).

```
Input: Given multi-view data $\mathbf{X}_v$, reduced dimension $d$,
  and sparsity level $s_v$;
Initialize: $(\mathbf{U}^0, \mathbf{P}_v^0)$;
While not converged do
  1:  Update $\mathbf{U}^{k+1}$ by
          $\min_{\mathbf{U}} \sum_{v=1}^{M} \|\mathbf{U} - \mathbf{X}_v\mathbf{P}_v^k\|_F^2$
          s.t. $\mathbf{U}^T\mathbf{U} = \mathbf{I}_d$;        (2)
  2:  Update $\mathbf{P}_v^{k+1} (v = 1, \ldots, M)$ by
          $\min_{\mathbf{P}_v} \sum_{v=1}^{M} \|\mathbf{U}^{k+1} - \mathbf{X}_v\mathbf{P}_v\|_F^2$
          s.t. $\|\mathbf{P}_v\|_{2,0} \leq s_v$;        (3)
  3:  Check convergence;
End while
Output: $(\mathbf{U}^{k+1}, \mathbf{P}_v^{k+1})$.
```

Next, the solutions for subproblem (2) and subproblem (3) will be discussed one by one.

A. Updating U

Considering the case when $\mathbf{P}_v^k (v = 1, \ldots, M)$ are fixed, the resulting subproblem (2) for estimating the variable $\mathbf{U}$ can be achieved by maximizing

$$\sum_{v=1}^{M} \mathrm{Tr}(\mathbf{U}^T\mathbf{X}_v\mathbf{P}_v^k).$$

Following a similar line of arguments as in [33], it admits a closed-form solution which is given by

$$\mathbf{U}^{k+1} = \mathbf{Q}\mathbf{V}^T,$$

where $\mathbf{Q}, \mathbf{V}$ are obtained from the singular value decomposition (SVD) as $\sum_{v=1}^{M} \mathbf{X}_v\mathbf{P}_v^k = \mathbf{Q}\mathbf{\Lambda}\mathbf{V}^T$.

B. Updating $\mathbf{P}_v$

The NHTP method has suggested excellent performance in vector space because only a relative fraction of linear equation systems needs to be solved to update its Newton direction [30]. However, the application to matrix cases has not been studied. The difficulty lies in how to exploit the sparse structure of matrices $\mathbf{P}_v (v = 1, \ldots, M)$ to achieve low computational complexity with fewer iterations. What follows is a brief summary of the efficient optimization strategy.

For a fixed $\mathbf{U}^{k+1}$, denote $f(\mathbf{P}_v) := \|\mathbf{U}^{k+1} - \mathbf{X}_v\mathbf{P}_v\|_F^2$. Then its gradient $\nabla f(\mathbf{P}_v)$ and Hessian $\nabla^2 f(\mathbf{P}_v)$ are

$$\nabla f(\mathbf{P}_v) = 2\mathbf{X}_v^T(\mathbf{X}_v\mathbf{P}_v - \mathbf{U}^{k+1}) \in \mathbb{R}^{d_v\times d}$$

and

$$\nabla^2 f(\mathbf{P}_v) = 2\mathbf{I}_d \otimes \mathbf{X}_v^T\mathbf{X}_v \in \mathbb{R}^{d_v d\times d_v d},$$

where $\otimes$ is the Kronecker product. For ease of description, let $\mathcal{S} := \{\mathbf{P}_v : \|\mathbf{P}_v\|_{2,0} \leq s_v\}$ and the projection onto $\mathcal{S}$ is $\Pi_{\mathcal{S}}(\cdot)$, which sets all but the s_v largest (in the ℓ_2-norm) rows to zero. Hence, the α_v-stationary point of (3) can be given by

$$\mathbf{P}_v = \Pi_{\mathcal{S}}(\mathbf{P}_v - \alpha_v\nabla f(\mathbf{P}_v)),$$

where $\alpha_v > 0$. Denote $\mathbb{T}_{s_v}(\mathbf{P}_v, \alpha_v)$ be the index set that covers the indices of s_v largest components in magnitude of $\mathbf{P}_v - \alpha_v\nabla f(\mathbf{P}_v)$. For any given $T_v \in \mathbb{T}_{s_v}(\mathbf{P}_v, \alpha_v)$ and α_v-stationary point $\mathbf{P}_v$, it derives

$$\begin{aligned}\mathbf{0} &= \mathbf{P}_v - \Pi_{\mathcal{S}}(\mathbf{P}_v - \alpha_v\nabla f(\mathbf{P}_v))\\ &= \begin{pmatrix}(\mathbf{P}_v)_{T_v}\\ (\mathbf{P}_v)_{\overline{T}_v}\end{pmatrix} - \begin{pmatrix}(\mathbf{P}_v)_{T_v} - \alpha_v\nabla_{T_v} f(\mathbf{P}_v)\\ \mathbf{0}\end{pmatrix}\\ &= \begin{pmatrix}\alpha_v\nabla_{T_v} f(\mathbf{P}_v)\\ (\mathbf{P}_v)_{\overline{T}_v}\end{pmatrix},\end{aligned}$$

where $\nabla_{T_v} f(\mathbf{P}_v) := (\nabla f(\mathbf{P}_v))_{T_v} \in \mathbb{R}^{s_v\times d}$ is the submatrix consisting of rows indexed by the T_v, and $\overline{T}_v$ is the complementary set of T_v in $\{1, \ldots, d_v\}$. Define

$$F(\mathbf{P}_v; T_v) := \begin{pmatrix}\nabla_{T_v} f(\mathbf{P}_v)\\ (\mathbf{P}_v)_{\overline{T}_v}\end{pmatrix} = \mathbf{0}. \tag{4}$$

The existence of α_v-stationary point of subproblem (3) can be guaranteed by the Theorem 3 in [30], and the solution of subproblem (3) can be achieved by calculating (4).

Now, it turns to use a Newton-based algorithm for solving (4). More precisely, let $\mathbf{P}_v^k$ be the current point, first select $T_v^k \in \mathbb{T}_{s_v}(\mathbf{P}_v^k, \alpha_v^k)$ and then find the Newton direction $\mathbf{D}_v^k$ by solving the following linear equations

$$\nabla F(\mathbf{P}_v^k; T_v^k) vec(\mathbf{D}_v^k) = -vec(F(\mathbf{P}_v^k; T_v^k)),$$

where $vec(\mathbf{D}_v^k)$ is the column vector obtained by stacking the rows of matrix $\mathbf{D}_v^k$ on top of one another, and $\nabla F(\mathbf{P}_v^k; T_v^k)$ is the Jacobian of $F(\mathbf{P}_v^k; T_v^k)$ at $\mathbf{P}_v^k$, i.e.,

$$\nabla F(\mathbf{P}_v^k; T_v^k) = \begin{bmatrix}\nabla^2_{T_v^k T_v^k} f(\mathbf{P}_v^k) & \nabla^2_{T_v^k \overline{T}_v^k} f(\mathbf{P}_v^k)\\ \mathbf{0} & \mathbf{I}_{(d_v - s_v)d}\end{bmatrix}.$$

Taking $\mathbf{D}_v^k = \begin{pmatrix} (\mathbf{D}_v^k)_{T_v^k} \\ (\mathbf{D}_v^k)_{\overline{T}_v^k} \end{pmatrix}$, it is easy to see that

$$\begin{aligned}\nabla^2_{T_v^k T_v^k} f(\mathbf{P}_v^k) vec\left((\mathbf{D}_v^k)_{T_v^k}\right) &= \nabla^2_{T_v^k \overline{T}_v^k} f(\mathbf{P}_v^k) vec\left((\mathbf{P}_v^k)_{\overline{T}_v^k}\right) \\ &\quad - vec\left(\nabla_{T_v^k} f(\mathbf{P}_v^k)\right) \end{aligned} \tag{5}$$

and

$$(\mathbf{D}_v^k)_{\overline{T}_v^k} = -(\mathbf{P}_v^k)_{\overline{T}_v^k}. \tag{6}$$

Step 1: To find $\mathbf{D}_v^k$, a linear (5) with ds_v equations and ds_v variables needs to be solved. The non-singular of matrix $\nabla^2_{T_v^k T_v^k} f(\mathbf{P}_v^k)$ may not be guaranteed for any k. To deal with this, the gradient direction $(\mathbf{D}_v^k)_{T_v^k} = -\nabla_{T_v^k} f(\mathbf{P}_v^k)$ compensates the case when the $(\mathbf{D}_v^k)_{T_v^k}$ in (5) is unsolvable. In addition, together with the property of vectorization operator $(\mathbf{B}^T \otimes \mathbf{A}) vec(\mathbf{D}) = vec(\mathbf{ADB})$ and the Hessian of $f(\mathbf{P}_v^k)$, one has

$$\begin{aligned} &\nabla^2_{T_v^k \overline{T}_v^k} f(\mathbf{P}_v^k) vec\left((\mathbf{P}_v^k)_{\overline{T}_v^k}\right) \\ &\quad = 2\left(\mathbf{I}_d \otimes (\mathbf{X}_v)^T_{T_v^k} (\mathbf{X}_v)_{\overline{T}_v^k}\right) vec\left((\mathbf{P}_v^k)_{\overline{T}_v^k}\right) \\ &\quad = 2vec\left((\mathbf{X}_v)^T_{T_v^k} (\mathbf{X}_v)_{\overline{T}_v^k} (\mathbf{P}_v^k)_{\overline{T}_v^k}\right). \end{aligned}$$

In particular, for the case of $n > d$, the cost of computing matrix-vector product $\nabla^2_{T_v^k \overline{T}_v^k} f(\mathbf{P}_v^k) vec((\mathbf{P}_v^k)_{\overline{T}_v^k})$ for a given matrix $(\mathbf{P}_v^k)_{\overline{T}_v^k}$ has now reduced to

$$\mathcal{O}(nds_v(d_v - s_v)) \Rightarrow \mathcal{O}(d^2 s_v(d_v - s_v)).$$

Step 2: In order to ensure the feasibility, i.e., $\|\mathbf{P}_v\|_{2,0} \le s_v$, the standard rule for Amijio line search $\mathbf{P}_v^{k+1} = \mathbf{P}_v^k + \alpha_v^k \mathbf{D}_v^k$ is modified as $\mathbf{P}_v^{k+1} = \mathbf{P}_v^{k+1}(\alpha_v^k)$, where

$$\begin{aligned} \mathbf{P}_v^{k+1}(\alpha_v^k) :&= \begin{pmatrix} (\mathbf{P}_v^k)_{T_v^k} + \alpha_v^k (\mathbf{D}_v^k)_{T_v^k} \\ (\mathbf{P}_v^k)_{\overline{T}_v^k} + (\mathbf{D}_v^k)_{\overline{T}_v^k} \end{pmatrix} \\ &= \begin{pmatrix} (\mathbf{P}_v^k)_{T_v^k} + \alpha_v^k (\mathbf{D}_v^k)_{T_v^k} \\ \mathbf{0} \end{pmatrix}. \end{aligned}$$

Overall, the solution for solving subproblem (3) can be summarized in Algorithm 2, which performs well for both $n > d$ and $n \le d$. The core idea is to take full advantage of the sparse structure by only calculating a linear equation system with ds_v equations and ds_v variables, so as to realize easy implementation with low computational complexity.

Algorithm 2: An Improved NHTP Method For Solving (3).

```
Input: Given data $\mathbf{X}_v$, $\mathbf{U}$, reduced dimension $d$, sparsity
  level $s_v$ and parameters $\gamma, \beta_v \in (0,1), \sigma \in (0, 1/2)$;
Initialize: $\mathbf{P}_v^0, \alpha_v, T_v^0 \in \mathbb{T}_{s_v}(\mathbf{P}_v^0, \alpha_v)$;
While not converged do
  1: Update $\mathbf{D}^k$ by solving (5) and (6), if it is solvable and
       $\langle \nabla_{T_v^k} f(\mathbf{P}_v^k), (\mathbf{D}_v^k)_{T_v^k} \rangle \le -\gamma \|\mathbf{D}_v^k\|_F^2 + \|(\mathbf{P}_v^k)_{\overline{T}_v^k}\|^2 / 4\alpha_v$.
     Otherwise, update $\mathbf{D}_v^k$ by
       $(\mathbf{D}_v^k)_{T_v^k} = -\nabla_{T_v^k} f(\mathbf{P}_v^k),\ (\mathbf{D}_v^k)_{\overline{T}_v^k} = -(\mathbf{P}_v^k)_{\overline{T}_v^k}$;
  2: Update $\mathbf{P}_v^{k+1} = \mathbf{P}_v^{k+1}(\alpha_v^k)$ with chosen $\alpha_v^k := \beta_v^{\tau_v^k}$
     by finding the smallest integer $\tau_v^k = 0, 1, \cdots$ such
     that
       $f(\mathbf{P}_v^k(\beta_v^{\tau_v^k})) \le f(\mathbf{P}_v^k) + \sigma \beta_v^{\tau_v^k} \langle \nabla f(\mathbf{P}_v^k), \mathbf{D}_v^k \rangle$;
End while
Output: $\mathbf{P}_v^{k+1}$.
```

III. Numerical Results

This section demonstrates the effectiveness and efficiency of the proposed SCGCCA over the state-of-the-art methods, including GCCA [34], SGCCA [25], and EGCCA [23].

A. Case Study on the Simulated Dataset

Motivated by the recent work [9], three different views of data can be generated according to

$$\begin{aligned} &\mathbf{X}_1 = \mathbf{u}^T(\mathbf{v}_1 + \mathbf{e}_1),\ \mathbf{X}_2 = \mathbf{u}^T(\mathbf{v}_2 + \mathbf{e}_2), \\ &\mathbf{X}_3 = \mathbf{u}^T(\mathbf{v}_3 + \mathbf{e}_3), \end{aligned}$$

where $\mathbf{u} \in \mathbb{R}^{1 \times 1000i}$ is a random vector satisfying the standard normal distribution for all $i \in \mathbb{R}_+$, $\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3 \in \mathbb{R}^{1 \times 300j}$ are

$$\begin{aligned} \mathbf{v}_1 &= [\underbrace{1 \cdots 1}_{50j}\ \underbrace{-1 \cdots -1}_{50j}\ \underbrace{0 \cdots 0}_{200j}], \\ \mathbf{v}_2 &= [\underbrace{0 \cdots 0}_{200j}\ \underbrace{1 \cdots 1}_{50j}\ \underbrace{-1 \cdots -1}_{50j}], \\ \mathbf{v}_3 &= [\underbrace{1 \cdots 1}_{50j}\ \underbrace{0 \cdots 0}_{200j}\ \underbrace{-1 \cdots -1}_{50j}], \end{aligned}$$

for all $j \in \mathbb{R}_+$, and $\mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3 \in \mathbb{R}^{1 \times 300j}$ are three random noise matrices. It is worth noting that when there exists no noise, $\mathbf{X}_1, \mathbf{X}_2, \mathbf{X}_3$ have column-wise sparsity. Therefore, the resulting $\mathbf{P}_1, \mathbf{P}_2, \mathbf{P}_3$ can achieve row-wise sparsity, and further, the nonzero rows correspond to these nonzero columns.

For the purpose of fair comparisons, the sparse parameters are chosen by five-fold cross-validation. In Algorithm 2, set $\sigma = 10^{-8}$, $\beta_v = 0.5$ and $\gamma = \gamma_k$ with updating

$$\gamma_k = \begin{cases} 10^{-10}, & \text{if } (\mathbf{P}_v^k)_{\overline{T}_v^k} = \mathbf{0}, \\ 10^{-4}, & \text{if } (\mathbf{P}_v^k)_{\overline{T}_v^k} \neq \mathbf{0}. \end{cases}$$

All of these methods are terminated by reaching 200 iterations or checking that whether the relative change in successive iterates is less than 10^{-3}.

Table I shows the simulated results for $i = 1, 5, 10, 50, 100$ and $j = 1, 5, 10$. The first column lists the problem scale under different $(n; d_1; d_2; d_3)$ and the other three columns list the CPU time in seconds. Moreover, the best method is highlighted in boldface. Compared with GCCA, SGCCA, and EGCCA, the proposed SCGCCA always requires the least running time. With the increase of problem scales, the improvement is more convincing. In particular, for the case $(100,000; 3,000; 3,000; 3,000)$, the efficiency has gained nearly 150 times. This demonstrates that the proposed method delivers consistent and competitive performance across all dimension configurations. The reason lies in that it only needs less than 5 iterations; see Fig. 1.

TABLE I
TIME FOR THE SIMULATED DATASET

Problem Scale	GCCA	SGCCA	EGCCA	SCGCCA
(1,000;300;300;300)	0.04	0.04	0.05	**0.01**
(5,000;300;300;300)	0.23	0.28	0.31	**0.03**
(10,000;300;300;300)	0.40	0.41	0.41	**0.07**
(50,000;300;300;300)	2.32	2.27	2.48	**0.34**
(100,000;300;300;300)	4.58	4.35	4.90	**0.66**
(1,000;1,500;1,500;1,500)	0.42	0.40	0.45	**0.02**
(5,000;1,500;1,500;1,500)	1.35	1.16	1.27	**0.12**
(10,000;1,500;1,500;1,500)	2.63	2.24	2.58	**0.24**
(50,000;1,500;1,500;1,500)	13.21	10.56	11.71	**1.18**
(100,000;1,500;1,500;1,500)	26.60	22.53	24.09	**2.35**
(1,000;3,000;3,000;3,000)	1.53	1.58	1.75	**0.17**
(5,000;3,000;3,000;3,000)	3.92	3.49	3.83	**0.23**
(10,000;3,000;3,000;3,000)	6.87	5.65	6.04	**0.45**
(50,000;3,000;3,000;3,000)	32.02	23.18	28.96	**2.29**
(100,000;3,000;3,000;3,000)	667.69	629.54	699.11	**4.91**

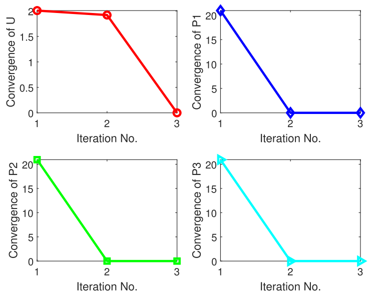
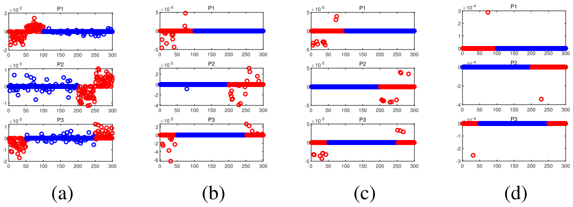


Fig. 1. Illustration of convergence.



(a) (b) (c) (d)

Fig. 2. Features extracted by different methods.

Good multi-view learning methods should obtain $\mathbf{P}_v$ that can identify correlated variables. Fig. 2 plots the extracted features by (a) SGCCA with $\lambda = 0.1$, (b) SGCCA with $\lambda = 1$, (c) SCGCCA with $s_1 = s_2 = s_3 = 10$, (d) SCGCCA with $s_1 = s_2 = s_3 = 1$. It can be observed that SGCCA can obtain sparsity and identify the correlation to some degree. When parameter values increase, the sparsity becomes strong, which means that these parameters should be chosen carefully; see Fig. 2(a) and Fig. 2(b). Moreover, the proposed SCGCCA can achieve different sparsity by tuning s_1, s_2, s_3. If only one variable is determined, just set $s_1 = s_2 = s_3 = 1$. To sum up, by introducing $\ell_{2,0}$-norm constrained optimization, SCGCCA is able to remove the redundant features and filter out noises at the same time. Furthermore, compared with the existing SGCCA, the performance is more robust.

B. Case Study on the MNIST Dataset

According to [35], the MNIST dataset collects 10-class 28 × 28 grayscale handwritten digits, and each class has 7,000 images. In this study, the dataset is divided into a training set, a tuning set, and a testing set, in which the training set has 50,000 images, the tuning set has 10,000 images, and the testing set has 10,000 images. Moreover, the dataset is rescaled and randomly rotated based on [36].

TABLE II
THE PERFORMANCE FOR MNIST DATASET

Performance	GCCA	SGCCA	EGCCA	SCGCCA
ACC (%)	74.37	78.82	79.13	**82.85**
Error (%)	25.63	21.18	20.74	**15.90**
Time (seconds)	18.67	19.34	19.88	**6.47**

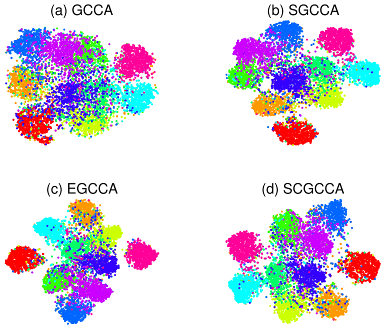


Fig. 3. The t-SNE embedding performance.

To evaluate the performance of multi-view learning, images are clustered into 10 classes and determined how well the clusters agree with ground-truth labels. The spectral clustering technique [37] is used to characterize the possible cluster shapes. Specifically, the binary weighting scheme is first adopted to construct a k-nearest-neighbor graph, and then the eigenvectors of the normalized graph Laplacian are applied to embed these samples into $\mathbb{R}^{10}$, and finally the K-means is run in the embedding to obtain the hard partition of the samples. Table II lists the clustering performance, where ACC denotes the clustering accuracy (%) [38] and Error denotes the classification error (%) [39]. It can be found that the proposed SCGCCA performs better than GCCA, SGCCA, and EGCCA in terms of ACC and Error. For example, the ACC of SCGCCA is 4.03% higher than that of SGCCA. Although GCCA, SGCCA, and EGCCA can complete the clustering task, a longer time is needed. However, the proposed SCGCCA can be solved efficiently and the time is reduced by 2 times.

To verify the clustering performance achieved by the compared methods, the visual results are quantified by embedding the projected features in 2D using t-SNE [40]; see Fig. 3. Although all of them fail to separate these classes completely (probably because the input variations are too complex to be captured by linear mappings), the proposed SCGCCA has a very outstanding class separation performance, which agrees with the relative clustering performances in Table II.

IV. CONCLUSION

In this letter, a sparsity constrained GCCA model was first considered, which incorporated GCCA with the $\ell_{2,0}$-norm for feature selection. Then, an optimization algorithm by combining the alternating minimization and Newton hard thresholding pursuit technique was developed. Finally, the performance was illustrated by sufficient experiments. The theoretical novelty is to introduce the $\ell_{2,0}$-norm constrained optimization, which is more faster and efficient than the relaxed $\ell_{2,1}$-norm regularizer. The practical importance is to improve the multi-view representation. In the future, extensions to the kernel- and deep neural networks-based GCCA need to be investigated.

REFERENCES

[1] 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.

[2] S. Sun, L. Mao, Z. Dong, and L. Wu, *Multiview Maching Learning*, Singapore: Springer, 2019.

[3] A. Blum and T. Mitchell, "Combining labeled and unlabeled data with co-training," in *Proc. 11th Annu. Conf. Comput. Learn. Theory*, 1998, pp. 92–100.

[4] F. R. Bach, G. R. Lanckriet, and M. I. Jordan, "Multiple kernel learning, conic duality, and the SMO algorithm," in *Proc. 21st Int. Conf. Mach. Learn.*, 2004, p. 6.

[5] L. Zhao, T. Zhao, T. Sun, Z. Liu, and Z. Chen, "Multi-view robust feature learning for data clustering," *IEEE Signal Process. Lett.*, vol. 27, pp. 1750–1754, 2020, doi: 10.1109/LSP.2020.3026943.

[6] F. De La Torre and M. J. Black, "A framework for robust subspace learning," *Int. J. Comput. Vis.*, vol. 54, no. 1, pp. 117–142, 2003.

[7] J. Wan and F. Zhu, "Cost-sensitive canonical correlation analysis for semi-supervised multi-view learning," *IEEE Signal Process. Lett.*, vol. 27, pp. 1330–1334, 2020, doi: 10.1109/LSP.2020.3010167.

[8] M. Sørensen, C. I. Kanatsoulis, and N. D. Sidiropoulos, "Generalized canonical correlation analysis: A subspace intersection approach," *IEEE Trans. Signal Process.*, vol. 69, pp. 2452–2467, 2021, doi: 10.1109/TSP.2021.3061218.

[9] J. Cai, W. Dan, and X. Zhang, "ℓ_0-based sparse canonical correlation analysis with application to cross-language document retrieval," *Neurocomputing*, vol. 329, pp. 32–45, 2019.

[10] Y. Yamanishi, J.-P. Vert, A. Nakaya, and M. Kanehisa, "Extraction of correlated gene clusters from multiple genomic data by generalized kernel canonical correlation analysis," *Bioinformatics*, vol. 19, pp. i323–i330, 2003.

[11] L. Du *et al.*, "Identifying progressive imaging genetic patterns via multi-task sparse canonical correlation analysis: A longitudinal study of the adni cohort," *Bioinformatics*, vol. 35, no. 14, pp. i474–i483, 2019.

[12] X. Yang, W. Liu, W. Liu, and D. Tao, "A survey on canonical correlation analysis," *IEEE Trans. Knowl. Data Eng.*, vol. 33, no. 6, pp. 2349–2368, Jun. 2021.

[13] X. Fu *et al.*, "Efficient and distributed generalized canonical correlation analysis for big multiview data," *IEEE Trans. Knowl. Data Eng.*, vol. 31, no. 12, pp. 2304–2318, Dec. 2019.

[14] N. Xu, J. Sun, J. Liu, and X. Xiu, "A novel scheme for multivariate statistical fault detection with application to the Tennessee Eastman process," *Math. Found. Comput.*, vol. 4, no. 3, p. 167, 2021.

[15] Y. Zhang, T. Wang, K. Liu, B. Zhang, and L. Chen, "Recent advances of single-object tracking methods: A brief survey," *Neurocomputing*, vol. 455, pp. 1–11, 2021.

[16] J. Chen, G. Wang, and G. Giannakis, "Graph multiview canonical correlation analysis," *IEEE Trans. Signal Process.*, vol. 67, no. 11, pp. 2826–2838, Jun. 2019.

[17] L. Gao, L. Qi, E. Chen, and L. Guan, "Discriminative multiple canonical correlation analysis for information fusion," *IEEE Trans. Image Process.*, vol. 27, no. 4, pp. 1951–1965, Apr. 2018.

[18] X. Chen, L. Han, and J. Carbonell, "Structured sparse canonical correlation analysis," in *Proc. 15th Int. Conf. Artif. Intell. Statist.*, 2012, pp. 199–207.

[19] T. Melzer, M. Reiter, and H. Bischof, "Nonlinear feature extraction using generalized canonical correlation analysis," in *Proc. Int. Conf. Artif. Neural Netw*. 2001, pp. 353–360.

[20] A. Benton, H. Khayrallah, B. Gujral, D. A. Reisinger, S. Zhang, and R. Arora, "Deep generalized canonical correlation analysis," 2017, *arXiv:1702.02519*. [Online]. Available: https://arxiv.org/abs/1702.02519

[21] M. Xu, Z. Zhu, X. Zhang, Y. Zhao, and X. Li, "Canonical correlation analysis with $l_{2,1}$-norm for multiview data representation," *IEEE Trans. Cybern.*, vol. 50, no. 11, pp. 4772–4782, Nov. 2020.

[22] L. Du *et al.*, "Identifying associations among genomic, proteomic and imaging biomarkers via adaptive sparse multi-view canonical correlation analysis," *Med. Image Anal.*, vol. 70, 2021, Art. no. 102003.

[23] C. I. Kanatsoulis, X. Fu, N. D. Sidiropoulos, and M. Hong, "Structured SUMCOR multiview canonical correlation analysis for large-scale data," *IEEE Trans. Signal Process.*, vol. 67, no. 2, pp. 306–319, Jan. 2019.

[24] C. Gao, Z. Ma, and H. H. Zhou, "Sparse CCA: Adaptive estimation and computational barriers," *Ann. Statist.*, vol. 45, no. 5, pp. 2074–2101, 2017.

[25] J. Cai and J. Huo, "Sparse generalized canonical correlation analysis via linearized Bregman method," *Commun. Pure Appl. Anal.*, vol. 19, no. 8, p. 3933, 2020.

[26] X. Xiu, Y. Yang, L. Kong, and W. Liu, "Data-driven process monitoring using structured joint sparse canonical correlation analysis," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 68, no. 1, pp. 361–365, Jan. 2021.

[27] T. Bouwmans, A. Sobral, S. Javed, S. K. Jung, and E.-H. Zahzah, "Decomposition into low-rank plus additive matrices for background/foreground separation: A review for a comparative evaluation with a large-scale dataset," *Comput. Sci. Rev.*, vol. 23, pp. 1–71, 2017.

[28] T. Blumensath and M. Davies, "Iterative hard thresholding for compressed sensing," *Appl. Comput. Harmon. Anal.*, vol. 27, no. 3, pp. 265–274, 2009.

[29] X. Yuan, P. Li, and T. Zhang, "Gradient hard thresholding pursuit," *J. Mach. Learn. Res.*, vol. 18, no. 1, pp. 6027–6069, 2017.

[30] S. Zhou, N. Xiu, and H. Qi, "Global and quadratic convergence of Newton hard-thresholding pursuit," *J. Mach. Learn. Res.*, vol. 22, no. 12, pp. 1–45, 2021.

[31] R. Wang, N. Xiu, and C. Zhang, "Greedy projected gradient-Newton method for sparse logistic regression," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 31, no. 2, pp. 527–538, Feb. 2020.

[32] M. Fazel, T. K. Pong, D. Sun, and P. Tseng, "Hankel matrix rank minimization with applications to system identification and realization," *SIAM J. Matrix Anal. Appl.*, vol. 34, no. 3, pp. 946–977, 2013.

[33] H. Zou, T. Hastie, and R. Tibshirani, "Sparse principal component analysis," *J. Comput. Graph. Statist.*, vol. 15, no. 2, pp. 265–286, 2006.

[34] D. M. Witten, R. Tibshirani, and T. Hastie, "A penalized matrix decomposition, with applications to sparse principal components and canonical correlation analysis," *Biostatistics*, vol. 10, no. 3, pp. 515–534, 2009.

[35] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proc. IEEE*, vol. 86, no. 11, pp. 2278–2324, Nov. 1998.

[36] W. Wang, R. Arora, K. Livescu, and J. Bilmes, "On deep multi-view representation learning," in *Proc. Int. Conf. Mach. Learn.*, 2015, pp. 1083–1092.

[37] A. Y. Ng, M. I. Jordan, and Y. Weiss, "On spectral clustering: Analysis and an algorithm," in *Proc. Adv. Neural Inf. Process. Syst.*, 2002, pp. 849–856.

[38] D. Cai, X. He, and J. Han, "Document clustering using locality preserving indexing," *IEEE Trans. Knowl. Data Eng.*, vol. 17, no. 12, pp. 1624–1637, Dec. 2005.

[39] C.-C. Chang and C.-J. Lin, "LIBSVM: A library for support vector machines," *ACM Trans. Intell. Syst. Technol.*, vol. 2, no. 3, pp. 1–27, 2011.

[40] L. Van der Maaten and G. Hinton, "Visualizing data using t-SNE," *J. Mach. Learn. Res.*, vol. 9, no. 11, pp. 2579–2605, 2008.