

Joint Cross-view Heterogeneous Discriminative Subspace Learning via Low-rank Representation

Yu Ding¹, Ao Li¹, Kezheng Lin¹, Xin Liu¹

{xuedinger66@163.com(Yu Ding), dargonboy@126.com(Ao Li), link@hrbust.edu.cn(Kezheng Lin)}

1.School of Computer Science and Technology, Harbin University of Science and Technology, Harbin, China

Abstract. The cross-view data are very unforced to capture due to the fact that different viewpoints or data collected from different sensors are already very common in recent years. However, cross-view data from different views present a significant difference, that is, cross-view data from different categories but in the same view have a higher similarity than the same category but within different views. To solve this problem, we have developed a dual low-rank representation framework to unbind these interleaved structures in a learning space. In addition, we consider that each cross-view sample of the same category is from isomorphic and heterogeneous information of two interlaced structures. Hence, we propose a powerful joint cross-view heterogeneous subspace feature learning model. In addition, the subspace learned by our algorithm contains more useful information and is more adaptable to cross-view data.

Keywords: Multi-view discriminative analysis, Cross-view subspace learning, Low-rank representation.

1 Introduction

Cross-view data, which depicts the same object from multiple angles, and contains richer classification and identification information than traditional single-view data, has been a hot research topic in the past few years[1]-[4]. Because different views data often has some special information, the samples of different views may be located in independent spaces. This phenomenon makes retrieval and recognition of objects or people very difficult. Hence, the requirements for the application of cross-view data methods are increasing.

In the past decade, many cross-view methods have emerged and achieved satisfactory results. Most of these methods learn a view-specific projection, projecting all views into a common or invariant subspace, where different views are represented as common or invariant representations. The most classic method is Canonical Correlation Analysis (CCA)[5], which establishes the relationship between two different views by maximizing the correlation coefficient of the two views and projects it into a common shared subspace. However, CCA can only be used in the case of two views. In order to overcome this difficulty, Multi-view CCA (MCCA)[6] was proposed. MCCA obtains multiple view-specific transformations between multiple views by maximizing the correlation coefficients between multiple views. In [7], Multi-view Discriminant Analysis (MvDA) was proposed, which adds Fisher constraints to the multi-view projection subspace to allow different views data to be projected into a discriminative shared subspace.

These methods usually project different views data into the same common subspace[8]-[10], and we find that cross-view data presents two different but intertwined structures in the original high-dimensional space, for example, a sample from cross-view data has both a class structure representing its own semantic information and a view structure containing its view information. The heterogeneous information brought by the view structure can affect the performance of retrieval and recognition. Therefore, we need to reduce the variance of the view to decompose the two structures to obtain the cross-view discriminant subspace. In addition, the dual low-rank representation method has been widely used recently[11]-[13], which has the ability to resist noise and makes the learned subspace robust. However, they ignore the similarity information between different views, for example, different views of the same sample contain part of the same feature information and unique information for each view. Therefore, cross-view subspace learning should not only discriminate learning based on the manifold structure of data, but also use complementary information between different views.

In this paper, we propose a novel cross-view discriminative feature subspace learning model called Joint Cross-view Heterogeneous Subspace Learning (JCHSL). Our algorithm learns the low-dimensional feature subspace through the joint heterogeneous information of different views, and adds discriminant graph variance constraints to unlock the potential double structure of cross-view data. In summary, the main contributions of our algorithm are composed of the following three aspects: (1) We use a dual low-rank representation to describe the manifold structure of cross-view data in a high-dimensional subspace. By discriminating view variance constraints, complex data is projected into a low-dimensional subspace structure that is easy to identify and its class structure and view structure are decomposed into each other. (2) We have designed a joint view heterogeneous feature subspace constraint that can extract hidden view heterogeneous information from different view data.

The other chapters of this article are designed as follows. Section II introduces the basic theory of the algorithm. In section III, we illustrate our proposed model and its optimization methods. The performance of the model is demonstrated through experiments in section IV, and the conclusions can be found in section V.

2 Related works

In this section, we introduce the related works of our proposed method, which are two different representation methods.

2.1 Ridge regression

We assume that a set of data consists of n samples $X = [X_1, X_2, \dots, X_n]$, and the output corresponding to the sample set $Y = [Y_1, Y_2, \dots, Y_n]$. We need find a regression coefficient matrix W to transform and reduce feature information for data X to touch semantic dictionary. Then, we use label information to construct the loss function as follows:

$$\min_W \|Y - WX\|_2^2 + \lambda \|W\|_2^2 \quad (2.1)$$

Ridge regression is a biased regression method dedicated to collinear data analysis, which is essentially an improved linear regression method. The linear regression model is widely used as an empirical risk function in supervised learning. Because there is a collinear relationship between cross-view data, we use ridge regression to obtain more stable regression coefficients.

2.2 Low-rank representation

Giving a set of data $X = [X_1, X_2, \dots, X_k]$ from k subspaces, which can be composed of a linear combination of dictionary and sparse noise matrix as follows:

$$\begin{aligned} \min_{Z, E} \text{rank}(Z) + \lambda \|E\|_1 \\ \text{s.t. } X = AZ + E \end{aligned} \quad (2.2)$$

in which A is a low-rank dictionary of the data. Z is the newly learned affine matrix with low-rank constraint, and $\text{rank}(\cdot)$ represents the rank of the matrix. We assume the noisy data E is sparse and handle it by l_1 -norm constraint. λ is the balanced parameter between linear representation and noise. However, the traditional low-rank representation only has a certain effect on ordinary data. For cross-view data, the data from the same view in the learned subspace will be close to each other, greatly reducing the accuracy of recognition. Therefore, our model uses a dual low-rank representation framework to solve this problem.

3 The proposes algorithm

In this section, we first introduce the proposed algorithm called Joint Cross-view Heterogeneous Subspace Learning. Then, an feasible solution is obtained though iterative method as well as complexity analysis.

3.1 Notations

We assume that there is a set of cross-view data $X = [X_1, X_2]$ from two views. The samples of the i th view $X_i = [X_{i1}, X_{i2}, \dots, X_{ic}]$ is consisting of l classes, where $X_{ir} \in R^{d \times m_r}$ denotes that d is the original dimensionality of training data and m_{ir} is the number of samples from the i th view and the r th class ($m_i = \sum_r m_{ir}, m = \sum_i m_i$). We construct the class structure matrix $Z_c \in R^{m \times m}$ and the view structure matrix $Z_v \in R^{m \times m}$ to adapt cross-view data subspace structure. Furthermore, we use the error matrix $E \in R^{p \times m}$ to fit the noisy data.

As a kind of feature subspace learning method, it is common to learn a projection matrix $P \in R^{d \times p}$, where P is reduced dimensionality. In addition, we design three weight matrices $W_0, W_1, W_2 \in R^{p \times l}$, where l is the number of classes for different categories of sample data,

to quantify cross-view feature in order to increase generalization for projection subspace.

3.2 Dual low-rank discriminative subspace learning

To the cross-view data, the samples from different views but within the same class has a large divergence due to two interlaced structures, which are class structure and view structure. As we discussed in section II, we adopt class structure matrix Z_c and view structure matrix Z_v as two affine matrices via dual low-rank representation to describe the class structure and view structure of cross-view data as follows:

$$\begin{aligned} \min_{Z_c, Z_v, E} \text{rank}(Z_c) + \text{rank}(Z_v) + \lambda \|E\|_1 \\ \text{s.t. } X = A_c Z_c + A_v Z_v + E, \end{aligned} \quad (3.1)$$

in which $A_c \in R^{d \times m}$ and $A_v \in R^{d \times m}$ are the class structure dictionary and view structure dictionary of the samples, respectively. λ is the trade-off parameter. $\text{rank}(\bullet)$ is the rank of the matrix, and in order to solve the minimizing rank problem, we use $\|\bullet\|_*$ to replace equivalently. With equation (3.1), the two interwoven structures, the class structure and view structure, are constructed independently. However, due to the characteristics of cross-view data, these two structures cannot be decomposed clear by unsupervised methods.

In order to our model to achieve the desired effect, we design a supervised discriminant view variance constraint term to help class structure and view structure decomposition of the cross-view data. Then, we promote our model as follows:

$$\begin{aligned} \min_{P, Z_c, Z_v, E} \|Z_c\|_* + \|Z_v\|_* + \lambda \|E\|_1 + \alpha \zeta(P, Z_c, Z_v) \\ \text{s.t. } P^T X = P^T (A_c Z_c + A_v Z_v) + E, P^T P = I, \end{aligned} \quad (3.2)$$

where $\zeta(P, Z_c, Z_v)$ is the supervised regularization term and α is the balancing parameter. In addition, the orthogonal constraint $P^T P = I$ is imposed to eliminate trivial solution and reduce the redundancy. More specifically, we design regularization terms based on Fisher's principle as follows:

$$\zeta(P, Z_c, Z_v) = \text{tr} \left(P^T \left(A_c Z_c L_c Z_c^T A_c^T - \phi A_v Z_v L_v Z_v^T A_v^T \right) P \right) \quad (3.3)$$

in which L_c and L_v are the Laplacian operator of \mathcal{W}^c and \mathcal{W}^v . ϕ is the balance coefficient inside the regularization term as 1. The forms of \mathcal{W}^c and \mathcal{W}^v are as follows:

$$\begin{aligned}
\mathcal{W}_{i,j}^c &= \begin{cases} 1, & \text{if } x_i \in \Pi_{k_1}^c(x_j), \text{ and } l_i = l_j, \\ 0, & \text{otherwise} \end{cases} \\
\mathcal{W}_{i,j}^v &= \begin{cases} 1, & \text{if } x_i \in \Pi_{k_2}^v(x_j), \text{ but } l_i \neq l_j, \\ 0, & \text{otherwise} \end{cases}
\end{aligned} \tag{3.4}$$

in which l_i and l_j are the labels of sample x_i , x_j , respectively. $x_i \in \Pi_{k_1}^c(x_j)$ denotes x_i is k_1 the nearest adjacency of data x_j within the same class. $x_i \in \Pi_{k_2}^v(x_j)$ means is k_2 the nearest adjacency of the same view data x_j . We design a supervised regularized view variance term in order to make within-class samples close to each other and between-class samples far away from each other, thereby unlocking two intertwined structures of cross-view data.

3.3 Joint view heterogeneous feature subspace learning

In order to obtain heterogeneous information and isomorphic information between different view data, we designed a joint view heterogeneous constraint so that the low-dimensional subspace contains these useful information. Therefore, we propose the final model by joint view heterogeneous constraints as follows:

$$\begin{aligned}
\min_{\substack{P, Z_c, Z_v, E \\ W_0, W_1, W_2}} & \left\| (\varphi W_0 + (1-\varphi)W_1)^T P^T X_1 - Y_1 \right\|_F^2 + \left\| (\varphi W_0 + (1-\varphi)W_2)^T P^T X_2 - Y_2 \right\|_F^2 \\
& + \varepsilon \left(\|W_1\|_F^2 + \|W_2\|_F^2 \right) + \gamma \|W_0\|_F^2 + \|Z_c\|_* + \|Z_v\|_* + \lambda \|E\|_1 + \alpha \zeta (P, Z_c, Z_v) \\
s.t. & P^T X = P^T (A_c Z_c + A_v Z_v) + E, P^T P = I
\end{aligned} \tag{3.5}$$

where $Y_i = [Y^1, Y^2, \dots, Y^{m_i}]$ is a class labels matrix from i th view. The zero-mean vector $Y^j = [-1, -1, \dots, 1, \dots, 1]^T \in R^l$ represents the j th column of Y_i . We use a weight matrix W_0 to quantify the view-shared feature that is owned jointly by all different views. And the weight matrices W_1, W_2 are used to elicit the individual component. In addition, we use the Frobenius norm constraint weight matrix to make our algorithm generalizable and eliminate trivial solutions. Two parameters γ and ε can adjust the values of the isomorphic weight matrix and the heterogeneous weight matrix, respectively. We also use the parameter φ to control the proportion of isomorphic and heterogeneous information in the low-dimensional subspace.

3.4 Optimization

In order to simplify the matrix we introduce the auxiliary matrix M and use the original data X as its own class structure dictionary A_c and view structure dictionary A_v . Our objective function is rewritten as follow:

$$\begin{aligned} \min_{\substack{P, Z_c, Z_v, E \\ W_0, W_1, W_2}} & \left\| (\varphi W_0 + (1-\varphi)W_1)^T M^T X_1 - Y_1 \right\|_F^2 + \left\| (\varphi W_0 + (1-\varphi)W_2)^T M^T X_2 - Y_2 \right\|_F^2 \\ & + \varepsilon \left(\|W_1\|_F^2 + \|W_2\|_F^2 \right) + \gamma \|W_0\|_F^2 + \|Z_c\|_* + \|Z_v\|_* + \lambda \|E\|_1 + \alpha \zeta (P, Z_c, Z_v) \\ \text{s.t.} & P^T X = P^T X (Z_c + Z_v) + E, P^T P = I, P = M \end{aligned} \quad (3.6)$$

We choose Augmented Lagrangian Methods (ALM) to solve the optimization problem. However, our objective function is non-convex. We need use first order Taylor expansion to simplify the objective function, and then use the Alternating Direction Multiplier Method (ADMM) to solve it.

Firstly, we transform the equation (3.6) into a graceful Lagrangian form:

$$\begin{aligned} \min_{\substack{P, Z_c, Z_v, E, Q \\ M, W_0, W_1, W_2}} & \left\| (\varphi W_0 + (1-\varphi)W_1)^T M^T X_1 - Y_1 \right\|_F^2 + \left\| (\varphi W_0 + (1-\varphi)W_2)^T M^T X_2 - Y_2 \right\|_F^2 \\ & + \varepsilon \left(\|W_1\|_F^2 + \|W_2\|_F^2 \right) + \gamma \|W_0\|_F^2 + \|Z_c\|_* + \|Z_v\|_* + \lambda \|E\|_1 + \alpha \zeta (P, Z_c, Z_v) \\ & + Tr \left(Q^T (P^T X - P^T X (Z_c + Z_v) - E) \right) + \frac{\mu}{2} \|P^T X - P^T X (Z_c + Z_v) - E\|_F^2 \\ & + \beta \|P - M\|_F^2 \end{aligned} \quad (3.7)$$

where Q is the Lagrange multiplier and $\mu > 0$ and $\beta = 1$ are the regularization parameters. Then, the equation is rewritten to a quadratic form via merging some terms as follows:

$$\begin{aligned} \min_{\substack{P, Z_c, Z_v, E, Q \\ M, W_0, W_1, W_2}} & \left\| (\varphi W_0 + (1-\varphi)W_1)^T M^T X_1 - Y_1 \right\|_F^2 + \left\| (\varphi W_0 + (1-\varphi)W_2)^T M^T X_2 - Y_2 \right\|_F^2 \\ & + \varepsilon \left(\|W_1\|_F^2 + \|W_2\|_F^2 \right) + \gamma \|W_0\|_F^2 + \|Z_c\|_* + \|Z_v\|_* + \lambda \|E\|_1 \\ & + \eta (P, Z_c, Z_v, E, Q, \mu) - \frac{1}{\mu} \|Q\|_F^2 + \beta \|P - M\|_F^2 \end{aligned} \quad (3.8)$$

where $\eta(P, Z_c, Z_v, E, Q, \mu) = \alpha \zeta(P, Z_c, Z_v) + \frac{\mu}{2} \left\| P^T X - P^T X (Z_c + Z_v) - E + \frac{Q}{\mu} \right\|_F^2$.

Similar to the conventional ALM, variable $P, Z_c, Z_v, M, W_0, W_1, W_2$ and E cannot be addressed simultaneously, but they are solvable individually when fixing other variables. To solve each sub-problem, η is approximated by the first order Taylor expansion. We define the right-bottom of the variable plus t as the optimized solution at the t th time. Then, each sub-problem at the $t+1$ th time is as follows:

Updating Z_c :

$$Z_{c,t+1} = \min_{Z_c} \frac{1}{\mu\rho} \|Z_c\|_* + \frac{1}{2} \|Z_c - Z_{c,t} + \nabla_{Z_c} \eta\|_F^2 \quad (3.9)$$

where

$\nabla_{Z_c} \eta = 2\alpha X^T P_t P_t^T X Z_{c,t} L_c - Q_t^T P_t^T X - \mu X^T P_t (P_t^T X - P_t^T X (Z_{c,t} + Z_{v,t}) - E_t)$ and $\rho = \|P_t^T X\|_2^2$. This can be addressed by singular value thresholding effectively[14].

Updating Z_v :

$$Z_{v,t+1} = \min_{Z_v} \frac{1}{\mu\rho} \|Z_v\|_* + \frac{1}{2} \|Z_v - Z_{v,t} + \nabla_{Z_v} \eta\|_F^2 \quad (3.10)$$

where

$\nabla_{Z_v} \eta = -2\alpha X^T P_t P_t^T X Z_{v,t} L_v - Q_t^T P_t^T X - \mu X^T P_t (P_t^T X - P_t^T X (Z_{c,t+1} + Z_{v,t}) - E_t)$.

Equation(3.9) can be addressed in the same way to Equation(3.8).

Updating E :

$$E_{t+1} = \min_E \frac{\lambda}{\mu} \|E\|_1 + \frac{1}{2} \left\| E - \left(P_t^T (X - X (Z_{c,t+1} + Z_{v,t+1})) + \frac{Q_t}{\mu} \right) \right\|_F^2 \quad (3.11)$$

This can be addressed by shrinkage operator[15].

Updating P :

$$P_{t+1} = \left(2\alpha X Z_n X^T + \mu X_n X_n^T \right)^{-1} \left(\beta M + X_n \left(E - \frac{Q}{\mu} \right)^T \right) \quad (3.12)$$

where we define $Z_n = Z_{c,t+1} L_c Z_{c,t+1}^T - Z_{v,t+1} L_v Z_{v,t+1}^T$ and $X_n = X - X (Z_{c,t+1} + Z_{v,t+1})$ for simplicity.

Updating M :

$$M_{t+1} = \min_M \left\| (\varphi W_0 + (1-\varphi)W_1)^T M^T X_1 - Y_1 \right\|_F^2 + \left\| (\varphi W_0 + (1-\varphi)W_2)^T M^T X_2 - Y_2 \right\|_F^2 + \beta \|P - M\|_F^2 \quad (3.13)$$

It is difficult to solve the equation (3.13) with non-convex constraints directly on Euclidean space. We use a gradient based approach to optimize the problem on the Stiefel manifold[15].

Algorithm 1

Input: data matrices X_1, X_2, Y_1, Y_2 , parameters $\alpha, \lambda, \varepsilon, \varphi, \gamma$

Initialize: $\theta = 10^{-6}, \psi = 1.1, \mu = 0.1, \mu_{\max} = 10^6, \beta = 1, t = 0, t_{\max} = 10^2$

while not converged or $t \leq t_{\max}$ do

1. Optimize $Z_{c,t+1}$ according to (3.9) by fixing other parameters;
2. Optimize $Z_{v,t+1}$ according to (3.10) by fixing other parameters;
3. Optimize E_{t+1} according to (3.11) by fixing other parameters;
4. Optimize P_{t+1} according to (3.12) by fixing other parameters;
5. Optimize M_{t+1} according to (3.13) by fixing other parameters;
6. Optimize $W_{0,t+1}$ according to (3.14) by fixing other parameters;
7. Optimize $W_{1,t+1}$ according to (3.15) by fixing other parameters;
8. Optimize $W_{2,t+1}$ according to (3.16) by fixing other parameters;
9. $P_{t+1} \leftarrow \text{orthogonal}(P_{t+1})$;
10. Optimize the multiplier Q_{t+1} ,

$$Q_{t+1} = Q_t + \mu \left(P_{t+1}^T \left(X - X \left(Z_{c,t+1} + Z_{v,t+1} \right) \right) - E_{t+1} \right);$$

11. Update the parameter μ by $\mu = \min(\mu_{\max}, \psi\mu)$;

12. Check the convergence conditions $\left\| P_{t+1}^T \left(X - X \left(Z_{c,t+1} + Z_{v,t+1} \right) \right) - E_{t+1} \right\|_{\infty} < \theta$;

13. $t = t + 1$.

end while

Output: $P, M, Z_c, Z_v, E, W_0, W_1, W_2$

Updating W_0 :

$$W_{0,t+1} = \varphi \left(\varphi^2 M^T X_1 X_1^T M + \varphi^2 M^T X_2 X_2^T M + \varepsilon I \right)^{-1} \left(M^T X_1 \left(Y_1^T - (1-\varphi) X_1^T M W_1 \right) + M^T X_2 \left(Y_2^T - (1-\varphi) X_2^T M W_2 \right) \right) \quad (3.14)$$

Updating W_1 :

$$W_{1,t+1} = (1-\varphi) \left((1-\varphi)^2 M^T X_1 X_1^T M + \varepsilon I \right)^{-1} M^T X_1 \left(Y_1^T - \varphi X_1^T M W_0 \right) \quad (3.15)$$

Updating W_2 :

$$W_{2,t+1} = (1-\varphi) \left((1-\varphi)^2 M^T X_2 X_2^T M + \varepsilon I \right)^{-1} M^T X_2 \left(Y_2^T - \varphi X_2^T M W_0 \right) \quad (3.16)$$

Updating Q :

$$Q_{t+1} = Q_t + \mu \left(P_{t+1}^T \left(X - X \left(Z_{c,t+1} + Z_{v,t+1} \right) \right) - E_{t+1} \right) \quad (3.17)$$

Finally, the detail process of optimization is listed in **Algorithm 1**. We set the parameters $\mu, \mu_{\max}, \theta, \psi, t_{\max}$ and tune the trade-off parameters $\alpha, \lambda, \varepsilon, \varphi, \gamma$ by the experiment. And we initialize the matrices $P, M, Z_c, Z_v, E, W_1, W_2$ at random.

4 Experiments results

In this section, we evaluate the performance of our algorithm on four standard data sets.

4.1 Datasets introduction and experimental setting

The **CMU-PIE** Face database contains 68 different people, and each category of cross-view face samples have 21 different illumination conditions and 9 different poses. We crop the face images to 64×64 size. We adopt 5 poses P05, P09, P14, P27, P29 and randomly divide the samples in each pose set into a test set and a training set. To 5 poses of the CMU-PIE faces, we divided each 2 poses into same group, where Case1: {P05, P09}, Case2: {P05, P14}, Case3: {P05, P27}, Case4: {P05, P29}, Case5: {P09, P14}, Case6: {P09, P27}, Case7: {P09, P29}, Case8: {P14, P27}, Case9: {P14, P29}, Case10: {P27, P29}.

The **COIL-100** object database includes 100 objects, a total of 7200 images, each object obtained 72 images and these images are caught with 5 degree rotation. The object images are cropped to 32×32 size and divided to two subsets as ‘‘C1’’ and ‘‘C2’’. In addition, C1 contains

the images in two point of view $V1 [0^\circ, 85^\circ]$ and $V2 [185^\circ, 265^\circ]$. Similarly, C2 obtains the images in $V3 [90^\circ, 175^\circ]$ and $V4 [270^\circ, 355^\circ]$.

The **Extended YaleB** face database consists of 16128 images, under 28 years old, including 64 images in different illumination conditions and 9 postures. We crop the face images to 32×32 size. We divide each person's images into four poses P1, P2, P3, P4 through experiments, which are approximately positive, so that samples can be more relevant to our experiments. We further partition each two poses into one group, where $V1 [P1, P2]$ and $V2 [P3, P4]$.

4.2 Experimental results

In order to evaluate the performance of our algorithm, we have selected several classic methods for comparison, such as PCA, LDA, LPP, LatLRR, LRCS, SRRS, MvDA, RMSL. In addition, we use the k-Nearest Neighbor classifier for performance evaluation of the extracted feature information by each method. To 5 poses of the CMU-PIE faces, the results of our algorithm and comparative experiment are shown in Tables 1&2. To COIL-100 objects, we select one from $V1$ and one from $V2$ as training set from each set of perspectives, and another

Table 1. Average recognition rates (%) of comparison methods on the CMU-PIE face dataset (Case1-5)

Methods	Case1	Case2	Case3	Case4	Case5
PCA	48.81±0.73	50.89±1.00	50.50±1.14	49.07±1.04	48.36±1.33
LDA	62.48±0.78	66.31±1.33	66.76±1.67	62.16±1.44	61.27±1.35
LPP	62.40±1.08	59.25±0.02	60.17±0.03	61.97±0.14	65.72±0.11
LatLRR	65.07±1.00	65.36±1.75	66.61±1.23	62.47±1.8	63.65±3.11
LRCS	95.68±1.01	91.83±0.64	92.30±0.59	95.48±0.43	89.60±1.08
SRRS	85.31±1.02	82.04±2.12	82.33±1.03	85.22±1.08	83.37±1.12
MvDA	95.71±2.2	92.02±0.9	91.5±1.76	95.42±0.18	90.62±0.22
RMSL	97.14±0.02	92.97±0.01	93.70±0.07	97.26±0.17	91.85±0.04
Ours	98.39±0.03	93.82±0.04	94.50±0.05	98.19±0.11	92.47±0.12

Table 2. Average recognition rates (%) of comparison methods on the CMU-PIE face dataset (Case6-10)

Methods	Case6	Case7	Case8	Case9	Case10
PCA	48.43±0.73	45.51±1.62	55.28±1.37	48.64±1.76	49.68±0.86
LDA	61.50±1.49	56.54±1.05	66.96±1.32	61.10±1.56	61.83±1.48
LPP	66.13±0.65	63.34±0.05	59.29±0.14	58.10±0.01	63.72±0.13
LatLRR	63.09±0.97	61.04±1.39	66.1±1.84	60.78±2.01	60.42±1.26
LRCS	89.23±0.62	95.57±1.02	87.42±0.54	90.88±0.92	90.64±0.71
SRRS	86.17±0.44	82.89±0.32	77.45±0.64	81.64±0.78	82.18±0.83
MvDA	91.03±0.19	95.32±0.18	87.32±0.17	90.86±0.23	92.38±0.21
RMSL	92.99±0.11	97.55±0.08	88.47±0.01	92.02±0.12	92.36±0.06
Ours	93.95±0.08	98.35±0.04	89.48±0.07	92.19±0.01	93.89±0.15

perspective as a test set. There are four experimental groups to evaluate the performance of all algorithm, as shown in **Figure 1**. For extended YaleB faces, our experimental setup is similar to COIL-100 objects. The recognition rates of the algorithm are shown in **Figure 2**.

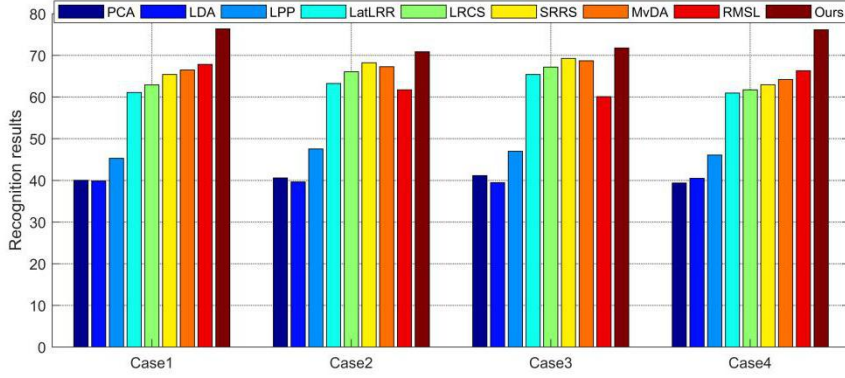


Fig. 1. Average recognition rates (%) of comparison methods on the COIL-100 object dataset

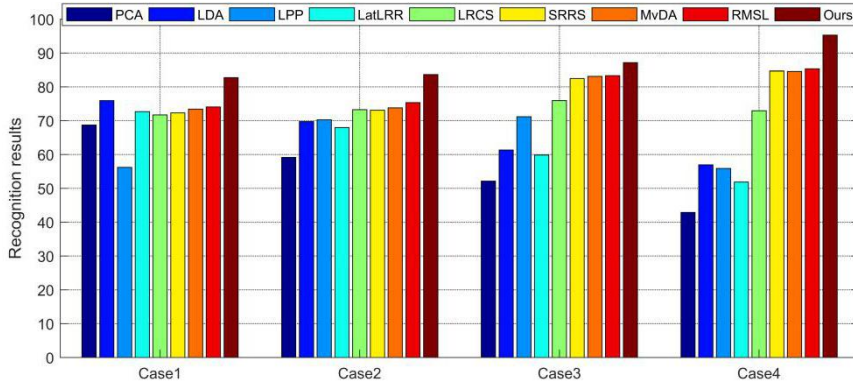


Fig. 2. Average recognition rates (%) of comparison methods on the extended YaleB face dataset

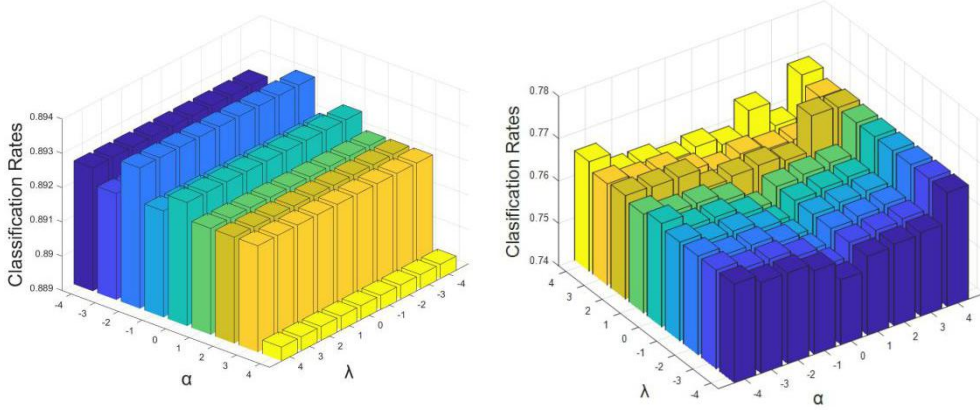
By comparing Table 1&2 and **Figure 1&2**, we can observe that our algorithm is more accurate than other traditional methods for cross-view data.

4.3 Sensitivity analysis of proposed algorithm

In this part, we will test the influences of the parameters and feature dimensionality selected by the algorithm on the recognition results.

Our algorithm has four parameters $\lambda, \alpha, \varepsilon, \gamma, \varphi$. We adjust parameters α and λ at the same time. We evaluate them respectively on CMU-PIE faces Case8 and COIL-100 Case1, and the **Figure 3** shows the results. In addition, we individually adjust parameters ε, γ and φ . We evaluate them on CMU-PIE faces Case8, and the **Figure 4** shows the results. Because parameters α and λ come from the same framework and parameters ε, γ and φ come from the joint cross-view heterogeneous subspace learning framework, they will affect each other in their own framework. From the **Figure 3&4**, the recognition rates are hardly sensitive to $\lambda, \varepsilon, \gamma, \varphi$. For the parameter α , the recognition rate has small fluctuations in a narrow

range. We can obtain almost consistent classification results in a wide range. The results points out that our algorithm is stable to parameter selection.



(a) CMU-PIE Case8

(b) Coil100 Case1

Fig. 3. The performance of our algorithm is evaluated on the four parameters influence $[\alpha, \lambda]$ on CMU-PIE faces Case8(a) and COIL-100 Case1(b), where the value from -4 to 4 denotes $[10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}, 1, 10, 10^2, 10^3, 10^4]$

Afterwards, we proof the dimensionality influence of our method in CMU-PIE faces Case8 and Coil100 Case1, and the Figure 5 shows the experiment results. From the results, algorithm performance is not sensitive to the dimensions. For the CMU-PIE faces Case8, classification performance increases slightly when the dimensionality goes up. Performance reach the highest around 300. In the Extend YaleB faces Case1, classification performance cuts down enough slightly with increasing of the dimensionality.

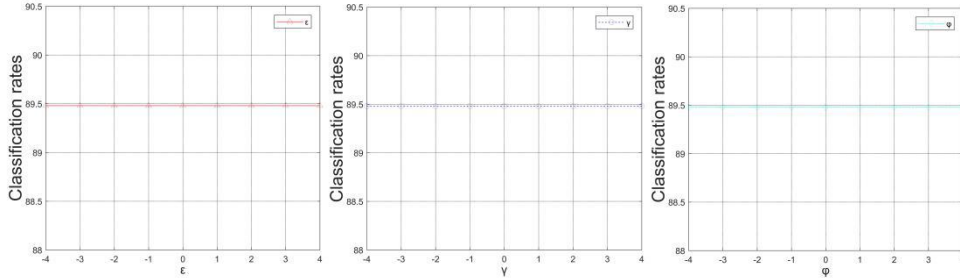


Fig. 4. The performance of our algorithm is evaluated on the four parameters influence $[\epsilon, \gamma, \phi]$ on CMU-PIE faces Case8, where the value from -4 to 4 denotes $[10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}, 1, 10, 10^2, 10^3, 10^4]$

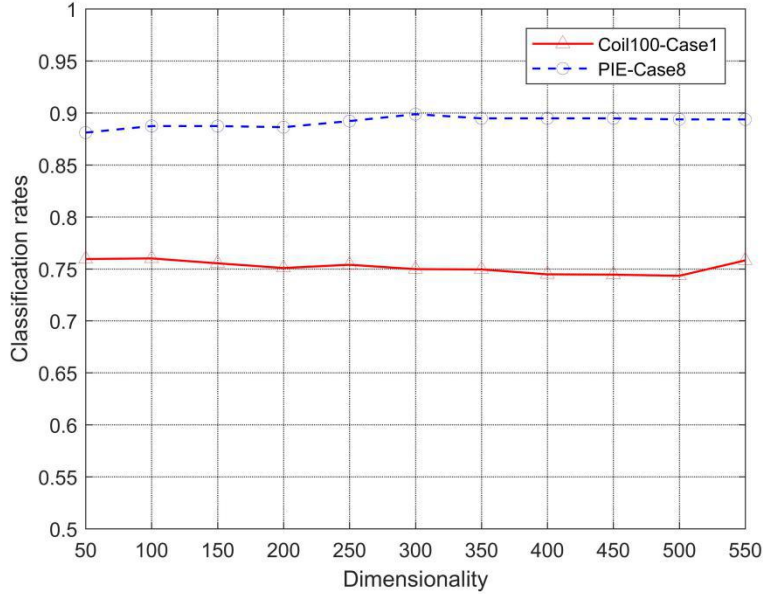


Fig. 5. Classification rates of different dimensionality on CMU-PIE faces

Case8 and Coil100 Case1.

5 Conclusion

To solve the difficulties caused by cross-view data, we propose a joint cross-view heterogeneous subspace learning method based low-rank constraint for image feature extraction and recognition tasks. In detail, we have established a dual low-rank representation framework to unlock the potential double structure of cross-view data through supervised view variance constraints. Also, we have established a joint cross-view heterogeneous subspace framework that combines the isomorphic and heterogeneous features of images from different perspectives to preserve more useful information for image recognition. At the same time, we propose a feasible solution to ensure convergence. Through experiments on three different public cross-view datasets, compared with other methods, our method has obtained obvious advantages.

Acknowledgment. This work was supported in part by the National Natural Science Foundation of China under Grant 61501147, in part by the University Nursing Program for Young Scholars with Creative Talents in Heilongjiang Province under Grant UNPYSCT-2018203, in part by the Natural Science Foundation of Heilongjiang Province under Grant YQ2019F011, in part by the Fundamental Research Foundation for University of Heilongjiang Province under Grant LGYC2018JQ013, and in part by the Postdoctoral Foundation of Heilongjiang Province under Grant LBH-Q19112.

References

- [1] Andrew, J. B. G. Arora, R. and Livescu, K.: Deep canonical correlation analysis. in International Conference on Machine Learning (2013)
- [2] Sharma, D. H. A. Kumar A. and Jacobs, D.: Generalized multiview analysis: A discriminative latent space, in IEEE Conference on Computer Vision and Pattern Recognition (2012)
- [3] Li, A. Wu, Z. Lu, H. Chen, D. and Sun, G.: Collaborative self-regression method with nonlinear feature based on multi-task learning for image classification. Vol. 6, pp. 43 513–43 525. IEEE Access (2018)
- [4] Zhang, H. F. P. Z. C. Hu, Q. and Cao,X.: Latent multi-view subspace clustering. pp. 4333–4341 (2017)
- [5] Hotelling, H.: Relations between two sets of variates. Vol. 28, nos. 3–4, pp. 321–377. Biometrika (1936)
- [6] Rupnik, J. and Shawe-Taylor, J.: Multi-view canonical correlation analysis. in Proc. Slovenian KDD Conf. Data Mining Data Warehouses. pp. 1–4 (2010)
- [7] Kan, M. Shan, S. Zhang, H. Lao, S. and Chen, X.: Multi-view discriminant analysis. Vol. 38, no. 1, pp. 188–194. IEEE Trans. Pattern Anal. Mach. Intell (2016)
- [8] Liu, S. Y. J. S. Y. Y. G. Lin, Z. and Ma, Y. : Robust recovery of subspace structures by low-rank representation. Vol. 35, no. 1, p. 171 1C184. IEEE Trans. Pattern Anal. Mach. Intell (2013)
- [9] Li, S. and Fu, Y.: Learning robust and discriminative subspace with lowrank constraints. Vol. 27, no. 11, pp. 2160–2173. IEEE Transactions on Neural Networks and Learning Systems (2016)
- [10] Li, A. Chen, D. Wu, Z. Sun, G. and Lin, K.: Self-supervised sparse coding scheme for image classification based on low rank representation. Vol. 13, no. 6, pp. 1–15. PLOS ONE (2018)
- [11] Ding, Z. and Fu, Y.: Robust multi-view subspace learning through dual low-rank decompositions. in Proc. 13th AAAI Conf. Artif. Intell. pp. 1181–1187 (2016)
- [12] Ding, Z. and Fu, Y.: Dual Low-Rank Decompositions for Robust Cross-View Learning. Vol. 28, no. 1, pp. 194-204. IEEE Transactions on Image Processing (2019)
- [13] Feng, Z. Yang, S. Wang, M. and Jiao, L.: Learning Dual Geometric Low-Rank Structure for Semisupervised Hyperspectral Image Classification. IEEE Transactions on Cybernetics (2019)
- [14] Cai, E. J. Candes, J.-F. and Shen, Z.: A singular value thresholding algorithm for matrix completion. Vol. 20, no. 4, p. 1956 1C1982. SIAM Journal on Optimization (2010)
- [15] Wen, Z. and Yin, W.: A feasible method for optimization with orthogonality constraints. Vol. 142, no. 1-2, p. 397 1C434. MP (2013)