Efficient Cross-Modal Retrieval Using Social Tag Information Towards Mobile Applications

Jianfeng He[1], Shuhui Wang[1(✉)], Qiang Qu[2], Weigang Zhang[3], and Qingming Huang[1]

[1] Key Lab of Intelligent Information Processing of Chinese Academy of Sciences (CAS), Institute of Computing Technology, CAS, Beijing 100190, China
jianfeng.he@vipl.ict.ac.cn, wangshuhui@ict.ac.cn, qmhuang@ucas.ac.cn

[2] The Global Center for Big Mobile Intelligence, Frontier Science and Technology Research Centre, Shenzhen Institutes of Advanced Technology, CAS, Shenzhen 518055, China
qiang@siat.ac.cn

[3] School of Computer Science and Technology, Harbin Institute of Technology, No. 2 West Wenhua Road, Weihai 26209, China
wgzhang@hit.edu.cn

Abstract. With the prevalence of mobile devices, millions of multimedia data represented as a combination of visual, aural and textual modalities, is produced every second. To facilitate better information retrieval on mobile devices, it becomes imperative to develop efficient models to retrieve heterogeneous content modalities using a specific query input, e.g., text-to-image or image-to-text retrieval. Unfortunately, previous works address the problem without considering the hardware constraints of the mobile devices. In this paper, we propose a novel method named Trigonal Partial Least Squares (TPLS) for the task of cross-modal retrieval on mobile devices. Specifically, TPLS works under the hardware constrains of mobile devices, i.e., limited memory size and no GPU acceleration. To take advantage of users' tags for model training, we take the label information provided by the users as the third modality. Then, any two modalities of texts, images and labels are used to build a Kernel PLS model. As a result, TPLS is a joint model of three Kernel PLS models, and a constraint to narrow the distance between label spaces of images and texts is proposed. To efficiently learn the model, we use stochastic parallel gradient descent (SGD) to accelerate the learning speed with reduced memory consumption. To show the effectiveness of TPLS, the experiments are conducted on popular cross-modal retrieval benchmark datasets, and competitive results have been obtained.

Keywords: Cross-modal retrieval · Multimedia
Partial least squares · Images and documents

C. Doulkeridis et al. (Eds.): MATES 2017, LNCS 10731, pp. 157–176, 2018.
https://doi.org/10.1007/978-3-319-73521-4_10

1 Introduction

With rapid advance in Internet technology and data device, amount of data is soaring exponentially. Among this, mobile data is one of main data source created by people. For instance, it is at least 5 hundred million images that are uploaded to the Internet everyday; it is around 20-h videos that are shared in each minute. Furthermore, mobile data tends to appear in the form of multimedia, such as texts, images, sounds and videos which are often applied to record the users' mood in Facebook, Twitter and other social application. It is also mentionable that the mobile data is often illustrated in two or more multimedia. Facebook and Twitter, meanwhile, are also strong evidence to support this point. A microblog of the Facebook is always finished through texts and images, and a video of the Twitter tends to include video, sound and even text. Thus, according to the current situation that heterogeneous modal data is used to describe a theme or one thing, the traditional content-based retrieval in single modality may not fulfill the users' requirement, and it is crucial and imperative to achieve cross-modal retrieval in mobile devices for users. Cross-modal retrieval is a newly proposed retrieval to use one modal query to retrieval the other one modal data [3,4,8,12,30,33]. To be specific, given a text query, then return content-related images; or given a music query, then return content-related video. An concrete example of cross-modal retrieval shown in the Fig. 1. However, the text features and the image features are in different feature space so that they can not be matched directly with each other. Hence, the key problem for cross-modal retrieval is achieving consistent feature representation for each heterogeneous modality [10,13–15,23,26–29,31,35].

Besides the above key problem, characteristics of mobile device, however, also bring more limitation when we apply the application of the model to the mobile devices. There are two main special points in mobile device: small memory (random access memory) and no GPU, thus it is unpractical to achieve the cross-modal retrieval by deep learning frameworks in mobile devices. The first contributing factor is that the deep learning asks for large memory which is at least several hundred megabytes to store its neural weights. The second contributing factor is that the GPU which does not exist in mobile devices is necessary to deep learning to accelerate calculating speed sharply. Then, though the weights of convolutional neural networks (CNN) can be stored in some high-performance mobile devices, it is slow for them to get the CNN features without the GPUs in the testing period. As a consequence of these two points, low hardware condition should also be taken into consideration when we design a model to solve cross-modal retrieval on mobile device. Specially, we put the training process of our model on a computer, and we care the efficiency and feasibility of testing process which has been done on the android virtual device (AVD) of Android Studio.

To learn the consistent feature representation based on low hardware condition, we just give up the deep learning framework but choose traditional subspace learning [1,2,5,9,10,15,22,32–34], which has been a common method to solve cross-modal retrieval and requires low hardware condition. To clarify our idea

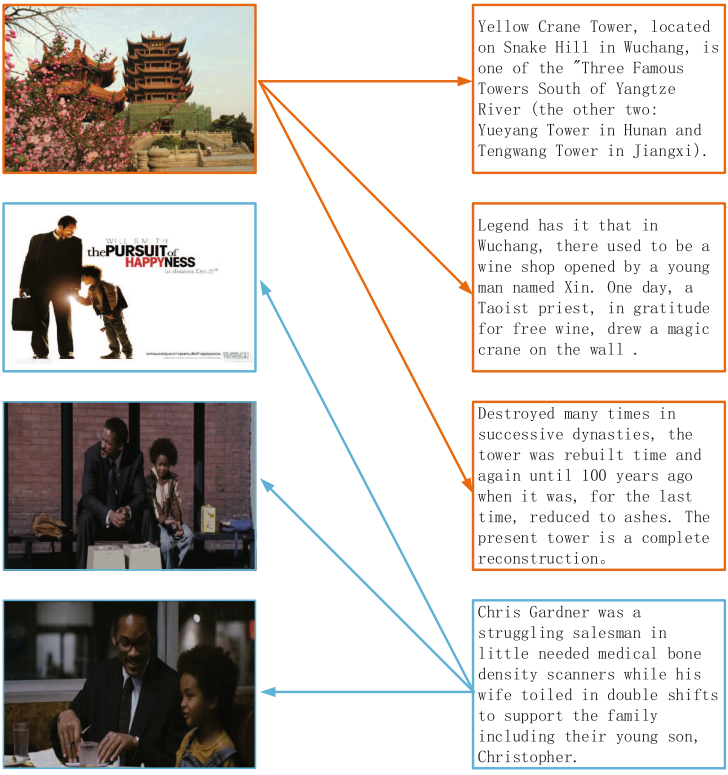


Fig. 1. The examples of cross-modal retrieval between the image and text. The orange rectangles, having common semantics in the Tower of Yellow Crane, show the image-to-text process: given an image, then return the related texts. As for the blue rectangles, which possess semantics in the movie, they illustrate the text-to-image process: return the related images according to the text. (Color figure online)

clearly, we use image-text retrieval as an example to describe our model, but the model is also suitable to other heterogeneous modalities.

The subspace learning methods learn a common feature space so as to match the image and text feature directly and preserve the correlations between image-text pairs. As one of the subspace learning methods, canonical correlation analysis (CCA) [7,15,19] projects the two modal features to a shared latent space which maximizes the correlations between two modalities. So far, many extensions of CCA have been used on similar area. In [15], the semantic correlation match (SCM) was proposed to get a semantic subspace by using logistic regressor based on CCA. In Verme [23] work, correlated semantic representation (CSR) was proposed to obtain a joint image-text representation and an unified formulation by learning a compatible function based on structural support vector machine (SVM). Besides, biliners model (BLM) [22] is also a kind of traditional

subspace learning methods, which gets rid of the diversity of between the different modal features.

Another traditional subspace learning method is partial least squares (PLS) [16,17] aiming at learning two latent spaces by maximizing the correlations between their latent variables. Sharma *et al.* [19] applied the PLS to build the relation between the latent variables of image and text in cross-media retrieval. In [10], PLS was applied into cross-media retireval. Besides solving the cross-modal retrieval problem, PLS also has been widely used in multi-view problem [11]. In Li *et al.* [11], PLS was used in cross-pose face recognition by constructing the relation between the coupled faces. In addition, PLS also has many extensions: In [21], bridge PLS was proposed by adding ridge-parameter in calculation to improve the efficiency in each iteration. Rosipal *et al.* [17] proposed the kernel PLS (KPLS) [17] by mapping the input variables into high dimension space so as to solve the nonlinear problem in linear space.

There is a common problem called semantic gap [15,25] existing in cross-modal retrieval. This problem is often solved by using label information due to its valuable semantic information [25]. In mobile devices, most users always give a tag to the microblogs or other things, such as the keywords given for each microblog, the items given by Amazon for each goods. The tags they provide equal label information in cross-modal retrieval. Then, by using the label information, the semantic gap decreases obviously. In Sharma's work [19], they proposed the framework named Generalized Multiview Analysis (GMA) to make use of labels for extracting multi-view features. Further, GMLDA and GMMFA, which are the application of GMA, shown competitive performances on the face recognition problem and cross-modal retrieval problem. In [10], Local Group based Consistent Feature Learning (LGCFL) was proposed which is a supervised joint feature learning method taking local group as priori.

Based on above discussion, we proposed a supervised algorithm, where the learned common feature space can be learned from two modalities. Our TPLS algorithm uses class indicator matrix indicating label information. At the same time, we introduce kernel partial least squares (KPLS) [17] to construct the relation between two multimedia modalities and the label modality. Because the KPLS can solve the nonlinear problem via linear method in its high dimension feature space, and it always gets better performance than PLS. In addition, we find that the common space constructed by label information are altered into two different spaces in KPLS iterative process, so we add novel constraint to minimize the divergence of label space. As a consequence of that, it makes the label space close to the other as possible as they can.

The remainder of the paper is organized as follows. In Sect. 2, we give a simple review of PLS and KPLS algorithms. Then, we will show our TPLS algorithm and its optimization in Sect. 3. In Sect. 4, experimental setting and result is shown. Finally, the conclusion is summarized in Sect. 5.

2 Preliminary

2.1 Partial Least Squares

PLS can construct the relation between two different modalities by maximizing the correlation between their latent variables. Let $X = [x_1, \cdots, x_n]^T$ represents the input variable with n training samples, where $x_i \in \mathbb{R}^{d1}$, $i = 1, \cdots, n$. Its latent variable represent as $V = [v_1, \cdots, v_n]^T \in \mathbb{R}^{n \times p}$, with $p \ll d1$. Respectively, let $Y = [y_1, \cdots, y_n]^T \in \mathbb{R}^{n \times d2}$ represent the output variable of training set. Its latent variables represents as $U = [u_1, \cdots, u_n]^T \in \mathbb{R}^{n \times p}$, with $p \ll d2$: PLS can be designed as:

$$\begin{cases} X = VP^T + \varepsilon_x \\ Y = UQ^T + \varepsilon_y \end{cases} \tag{1}$$

where the P and Q represent matrices of loadings, the ε_x and ε_y are the matrices of residuals. And by means of the low dimension latent variables V and U, we can further get a regression coefficient matrix $B \in \mathbb{R}^{d1 \times d2}$ to get the relation between X and Y:

$$B = X^T U (V^T X X^T U)^{-1} V^T Y \tag{2}$$

$$Y = XB^T + \varepsilon_B \tag{3}$$

where ε_B is the matrix of residuals.

PLS can be solved by traditional iterative algorithm calculating the first dominant eigenvector to get the weight vectors r, s. After i-th iteration, we can obtain the i-th latent vectors $v_i = Xr_i$, $u_i = Ys_i$ which respectively construct the i-dimension of latent variables V, U, and coefficient matrix B shown as Fig. 2. In traditional iterative algorithm, the object function can be described as follow:

$$[cov(v, u)]^2 = \max_{|r|=|s|=1} [cov(Xr, Ys)]^2 \tag{4}$$

Furthermore, according to the [16], we can also describe the object function of PLS as follow:

$$< v, u > = \arg\max_{r,s} < Xr, Ys > = \arg\max_{r,s} r^T X^T Y s \quad s.t. \quad r^T r = 1, s^T s = 1 \tag{5}$$

where $< a, b > = a^T b$ is the inner product of vector.

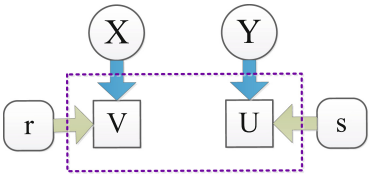


Fig. 2. The structure diagram of the PLS model.

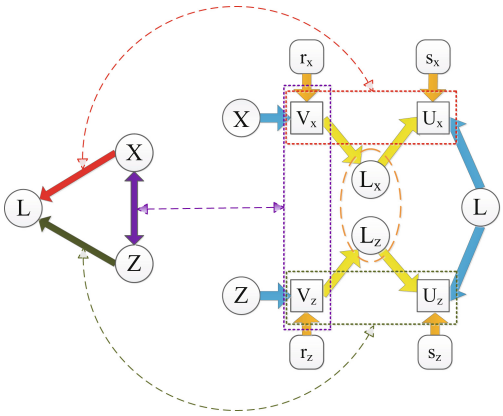


Fig. 3. The structure diagram of TPLS shows as above, in which the latent variables correlations constraint is indicated by three square dotted boxes, and Distance of target space constraint is indicated by the elliptical dotted boxes.

3 Trigonal Partial Least Squares

3.1 Trigonal Partial Least Squares Algorithm

Assume that there are two sets of multimedia kernel feature, $X = (x_1, x_2, \cdots, x_n)^T$, and $Z = (z_1, z_2, \cdots, z_n)^T$ from different modalities, where x_i is in n dimensions and z_i is also in n dimensions. Besides, the two sets of data are in c classes. And we construct the class indicator matrix $L = [l_1, l_2, \cdots, l_n]^T$, where l_i is a binary class indicator vector in c dimensions with all elements being zeros except for the place corresponding its classes. In other words, if sample z belongs to k-th class, then its class indictor vector l is $l_k = 1$ and $l_j = 0$ for $j \neq k$. As a result of two sets of data, we also have two class indicator matrices L_x, L_z respectively.

In our approach, we want to utilize label information to learn consistent feature representation. From Eq. 3, it is obvious that we need the regressor coefficient matrix B_x, B_y to map the input variables which are heterogenous features X, Z into the target feature spaces constructed by label information. So our approach learns four weight vectors r_x, s_x, r_z and s_z, which construct latent vectors $v_x = Xr_x$, $u_x = Xs_x$, $v_y = Vr_y$, $u_y = Vs_y$ respectively, and further construct the regressor coefficient matrices B_x, B_y via Eq. 2. And in our approach, we propose object function based three constraints as follow:

$$\arg \min_{r_x, r_z, s_x, s_z} F = G + D + P \tag{6}$$

$$s.t. \quad r_x^T r_x = 1, r_z^T r_z = 1, s_x^T s_x = 1, s_z^T s_z = 1$$

where G is a correlations constraint item defined on latent variables of heterogenous modal features and the class indicator matrix, D is a distance constraint

item of iterative output to reduce the difference between two modalities in each iterations. P is a transformation constraint item, which is also a regularization item defined on the weight vectors. The TPLS model is demonstrated in Fig. 3.

Latent variables correlations constraint: Previous work [19] uses PLS in cross-modal retrieval by setting one modal feature as input variable, and the other modal feature as output variable. This usage of PLS constructs the relation between two sets of modal feature, via maximizing their corresponding latent variables. Actually, label information is available for training the embedding [6]. More specifically, many heterogenous modal pairs own a unique label which is variant from each other. Taking account that heterogenous modalities share common labels, if we can establish the relation between two modalities via label in more direct way, it will help to improve the performance of cross-modal retrieval. Based on this consideration, we take the label as the output variables. In our method, label information is represented by class indicator matrices L_x, Lz. And then the label information is introduced in our method through PLS model as follow,

$$X = V_x P_x^T + \varepsilon_x \tag{7}$$

$$L_x = U_x Q_x^T + \varepsilon_{Lx} \tag{8}$$

$$Z = V_z P_z^T + \varepsilon_z \tag{9}$$

$$L_z = U_z Q_z^T + \varepsilon_{Lz} \tag{10}$$

In the above equations, two PLS model are established: Eqs. 7 and 8 are one PLS model between X and its class indicator matrix L_x, Eqs. 9 and 10 are the other one PLS model between Z and L_z. Therefore, according to the Eq. 5, we get latent variables correlations constraint of multimedia and label as follow,

$$G_1 = \lambda_1 < Xr_x, Ls_x > + \lambda_2 < Zr_z, Ls_z > \tag{11}$$

Besides above two PLS processes, we want that X modal latent variables V_x and text latent variables V_z can express data variability to the utmost extent, like the PCA,

$$\begin{cases} var(v_x) \to \max \\ var(v_z) \to \max \end{cases} \tag{12}$$

at the same time, we also ask v_x to explain v_z as possible as it can, based on CCA, the correlation between v_x and v_y should be maximized as follow,

$$cov(v_x, v_z) = \sqrt{var(v_x)var(v_z)}r(v_x v_z) \to \max \tag{13}$$

based on Eqs. (12) and (13), we can obtain our multimedia-multimedia correlation constraint as follow,

$$G_2 = \lambda_3 < Xr_x, Zr_z > \tag{14}$$

thus, we put multimedia-label correlation constraint G_1 and multimedia-multimedia correlation constraint G_2 together to obtain latent variables correlations constraint G as follow,

$$G = G_1 + G_2 \tag{15}$$

Via Eq. 15, firstly, we can map both two multimedia modal features X, Z to a common feature space constructed by label information in two PLS processes. Secondly, we maximize the correlation between latent variables of two modalities based the idea of PCA and CCA, which is also a PLS process.

Distance of iterative output constraint: By using Eqs. 7–10, we construct two PLS processes which set the class indicator matrices L_x and L_z as output variables respectively. It is deserved to point out that, the initial class indicator matrices L_x, L_z are same, but as a result of different initial condition in each PLS iteration, caused by respective different latent vectors v_x and v_z, the class indicator matrices L_x and L_z will be different in the subsequent iterative process shown as Eq. 19. The reason for that result is that two PLS processes have different input variables which are multimedia modal feature respectively. Considering that we should achieve the consistent feature representation for different modalities, so the output latent space of U_x and U_z should be in the same space. However, looking for two output latent variables U_x and U_z in a same space is a too strong constraint, which will destroy the PLS training process. So we add a soft constraint which is the distance constraint between L_x and L_z in iterative process. That is, we want to make the distance of L_x and L_z close to each other in each iterative process, thus the initial condition of each iterative process is as proximal as possible, and finally the output latent spaces U_x and U_z are close to each other in PLS training process. According to above analysis, we use the distance of iterative output as follow,

$$D = \lambda_4 \| L_x - X r_x r_x^T X^T L_x - (L_z - Z r_z r_z^T Z^T L_z) \|_F^2 \tag{16}$$

By adding the D, we use the Frobenius norm of matrix to reach the initial condition in each iteration proximal to the each other in each iteration.

Transformation constraint: This constraint item can be expressed as follow to prevent over fitting:

$$P = \frac{1}{2}(\|r_x\|^2 + \|r_z\|^2 + \|s_x\|^2 + \|s_z\|^2) \tag{17}$$

3.2 RBF Kernel and Deflation Detail

Specially, we construct TPLS based on KPLS and adopt RBF kernel to extract the kernel feature X and Z, the RBF kernel function can be described as follow:

$$k(x, x') = \exp(-\frac{||h - h'||^2}{2l_{rbf}^2}) + \sigma_w^2 \varepsilon_{h,h'} \tag{18}$$

where $2l_{rbf}^2$, σ_w^2 denote the parameters of the RBF bandwidth and the variance of noise respectively.

As a result of using kernel features X, Z, we can obtain the deflation of the matrices X, Z, and class indicator matrices L_x, L_z. The deflation of KPLS is different to PLS after extraction of the i-th latent vector v as follow,

$$\begin{cases} X^{i+1} = X^i - v_x v_x^T X^i - X^i v_x v_x^T + v_x v_x^T X^i v_x v_x^T \\ Z^{i+1} = Z^i - v_z v_z^T Z^i - Z^i v_z v_z^T + v_z v_z^T Z^i v_z v_z^T \\ L_x^{i+1} = L_x^i - v_x v_x^T L_x^i \\ L_z^{i+1} = L_z^i - v_z v_z^T L_z^z \end{cases} \tag{19}$$

3.3 Optimization

Then, we can obtain the optimal solution of TPLS by Stochastic Gradient Descent. And the gradient of each variable is solved as follow:

$$\begin{cases} \dfrac{\partial F}{\partial r_x} = \lambda_4 \dfrac{\partial D}{\partial r_x} - \lambda_1 X_z L s_x + \lambda_3 X_z Z r_z + r_x \\ \dfrac{\partial F}{\partial r_z} = \lambda_4 \dfrac{\partial D}{\partial r_z} - \lambda_2 Z_z L s_z + \lambda_3 Z_z X r_x + r_z \\ \dfrac{\partial F}{\partial s_x} = -\lambda_1 L X r_x + s_x \\ \dfrac{\partial F}{\partial s_z} = -\lambda_2 L Z r_z + s_z \end{cases} \tag{20}$$

where

$$\begin{cases} \dfrac{\partial D}{\partial r_x} = -2(X^T A L_x^T X r_x + X^T L_x A^T X r_x) \\ +2(X^T L_x L_x^T X r_x r_x^T X^T X r_x + X^T X r_x r_x^T X^T L_x L_x^T X r_x) \\ \dfrac{\partial D}{\partial r_z} = -2(Z^T B L_z^T Z r_z + Z^T L_T A^T Z r_z) \\ +2(Z^T L_T L_z^T Z r_T r_z^T Z^T Z r_z + Z^T Z r_T r_z^T Z^T L_T L_z^T Z r_z) \\ A = L_x - L_z + Z r_T r_z^T Z^T L_z \\ B = L_x - L_z - X r_x r_x^T X^T L_z \end{cases} \tag{21}$$

by using the SGD (Stochastic Gradient Descent), we only ask for a small part of training data in each iteration and then reduce the memory consumption in comparison with using BGD (Batch Gradient Desect) which makes usage of whole data set.

After we have solved the weight vectors r_x and r_z, we can further solved the latent vectors v_x and v_z. Similarly, we can get the latent vectors u_x and u_z. This was followed by solving the regression coefficient matrices B_x showing relation between between X and L and B_z showing relation between Z and L.

Algorithm 1. The algorithm of TPLS

Input:
the image kernel feature X, the text kernel feature Z, the class indicator matrices L_x and L_z, the dimension of latent variables c, the batch setting for SGD β

Output:
the weights matrices R^x and R^z, the latent variables V^x and V^z, the matrices of regression coefficients B^x and B^z, the residual matrices ε_B^X,ε_B^Z, and our regression models M^x,M^z

Initialize: $E_1^x = X$, $E_1^z = Z$, $F_1^x = L_x$, $F_1^z = L_z$
for $k = 1$ to c **do**
Calculate the k-th weight vectors r_k^x,r_k^z by stochastic gradient descent algorithm using Eqs. 20 and 21
Calculate the k-th latent vector:
$v_k^x = E_k^x r_k^x$, $v_k^z = E_k^z r_k^z$
Deflate E_k,F_k matrices using Eq. 19 respectively
end
Calculate the regression coefficient matrices using Eq. 2 respectively
Calculate the residual matrix:
$\varepsilon_B^x = F_k^x - E_k^x B^x$ $\varepsilon_B^z = F_k^z - E_k^z B^z$
Obtain the PLS regression models M^x,M^z using Eq. 3 respectively
return M^X, M^Z;

3.4 Computational Complexity Analysis

Lastly, we briefly analyze the computational complexity of TPLS method, which involves c iterations because of the c dimensions of latent variables, and each dimension is obtained by gradient descent algorithm, in which the max iterations set as z. Set n as the number of sample pairs in the training set, thus image feature and text feature are n dimensions as a result of RBF kernel. The computational complexity of TPLS is $O(czn^2)$.

3.5 Test TPLS on AVD

Above training process is done on a computer offline, after which, we test our model in an AVD, in which the configure setting is Device Nexus 5X API 26. And we just introduce how to test our TPLS on AVD in this section. The diagram of the how to apply TPLS in AVD is shown in Fig. 4.

As we have solved the B_x and B_z in the training process, we then store the two matrices and the feature of test set extracted in advance in the AVD. Then, according to the Fig. 4, our model can extract the image feature or the texts feature according to the query at first. The query feature is then project to the common space, which is the consistent representation in the picture. The parameters here are the B_x and B_z according to the query. After we have calculated consistent representation, we then get the similarity between query and respective dataset which is the feature of test set extracted in advance. This is followed by outputting the top ranked match results to the users. It is mentionable that we are not allowed to show the original images or texts, for these are too large to store in the mobile phone, such as the wiki dataset, which has only six hundred and ninety three images, requiring nearly 1 GB to store

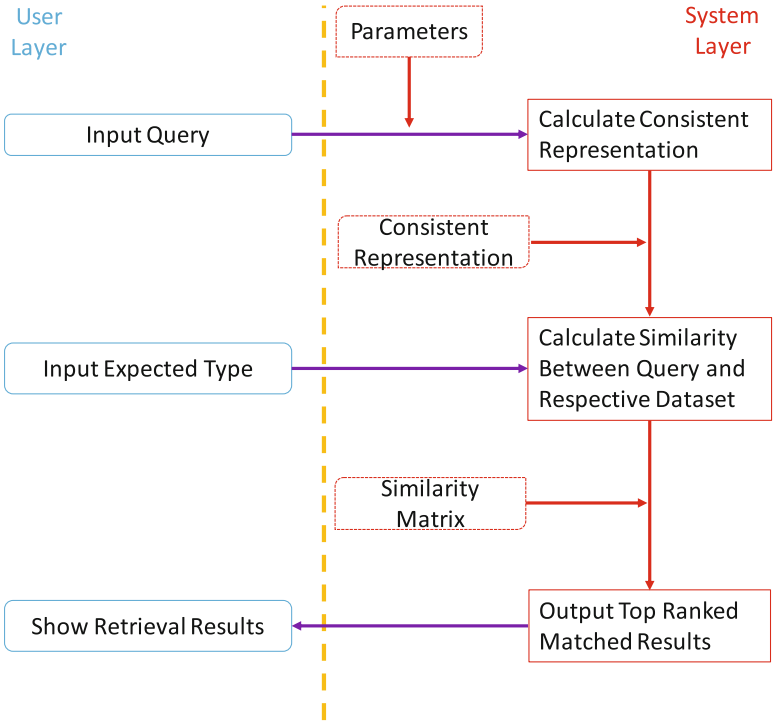


Fig. 4. The diagram of the how to apply TPLS in AVD.

them, let alone the texts. Hence, we just show the retrieval results in a form of their file name. Also, we show the demo of text-to-image task in the AVD in the Fig. 5, in which, we have two choices "IMAGE QUERY" and "TEXT QUERY" shown in the Fig. 5(a). After we click the button "TEXT QUERY", we step into the Fig. 5(b) which allows us to do the text-to-image task. Then, the texts have been keyed and we click the button "TEXT QUERY" again, the results are shown in the Fig. 5(c). Finally, we can click the triangle button to return the initial screen shown as the Fig. 5(a) which can do the new cross-modal retrieval.

4 Experiments

In this section, we test the proposed method on two popular databases to show its effectiveness.

4.1 Experimental Databases

Wiki is constructed from the Wikipedia including 2866 image-text pairs with 10 different categories. The images are represented by 128-dimensional vector based on SIFT descriptors and the texts are represented by 10-dimensional LDA

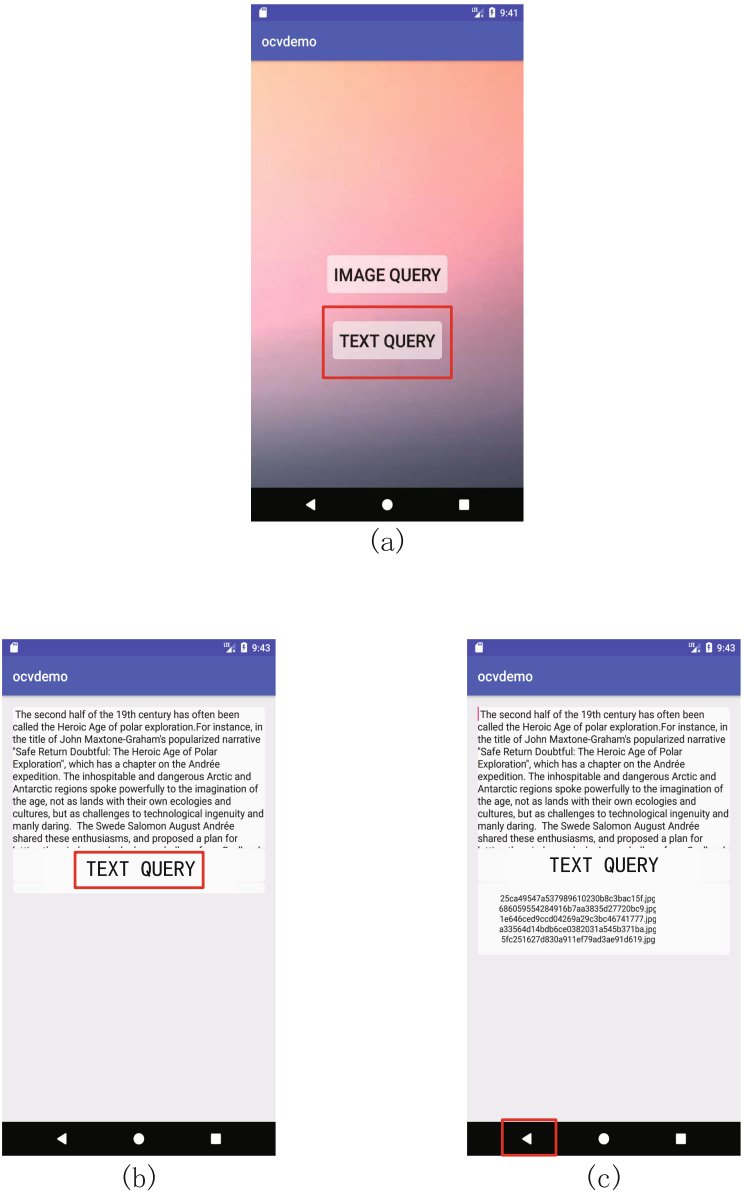


Fig. 5. The screenshots of the Demo in AVD in terms of the text-to-image task.

(latent Dirichlet allocation) feature. We randomly select 2173 image-text pairs for training set and 693 image-text pairs for testing set.

Flickr is a subset chosen from NUS-WIDE, which is crawled from the Flicker website. The Flickr database consists of 5730 single-label images associated with

their tag text. For feature representation, we use 500-dimensional bag-of-words based on SITF descriptors as image feature and 1000-dimensional word frequency based tag feature as text feature. The image-text pairs are selected from NUS-WIDE in which have top-10 largest numbers of images. As a consequence of that, we randomly choose 2106 image-text pairs as training set and the rest 3624 pairs as testing set.

4.2 Evaluation Metric

In our experiment, we use MAP (Mean Average Precision) and PR (Precision-Recall) curve to show the effectiveness of MDLL.

MAP has been widely used to evaluate the overall performance of cross-modal retrieval, such as [10,15,20,24,28]. To compute MAP, we first evaluate the average precision (AP) of a retrieved database including N retrieved samples by $AP = \frac{1}{T}\sum_{r=1}^{N} E(r)\delta(r)$, where T is the number of the relevant samples in the retrieved database, $E(r)$ denotes the precision of the top r retrieved samples, and $\delta(r)$ is set to 1 if the r-th retrieved sample is relevant (on above three databases, a retrieved sample is relevant if it shares at least one label with the query) and $\delta(r)$ is 0 otherwise. Then by averaging the AP values over all the queries, MAP can be calculated.

Besides, PR curve is a classical measure of information retrieval or classified performance. Assume that the set S_1 includes the samples in which real labels are denoted by L_r. The classifier picks out the set S_2 samples in which labels are classified into L_r. In the set S_2, the samples in which real labels are L_r construct the set S_3. Thus, we can calculate the precision ratio: $PR = \frac{|S_3|}{|S_2|}$ and the recall ratio: $RR = \frac{|S_3|}{|S_1|}$, where $|A|$ means the number of elements in set A. Furthermore, we get different PR-RR values via the different classified setting and then draw precision-recall curve in which the vertical coordinate is precision ratio and the horizontal coordinate is recall ratio.

4.3 Compared Scheme

We compare our approach with PLS, Kernel PLS (KPLS), Semantic Correlation Matching (SCM), Correlated Semantic Representation (CSR), Generalized Mutiview Marginal Fisher Analysis (GMMFA) and Generalized Multiview LDA (GMLDA) in two retrieval tasks. In PLS, the modal latent variables are obtained by maximize the correlation between the latent variables of images and texts. KPLS maps the original feature into a high dimension feature space so as to construct the linear model to solve the nonlinear problem. As for SCM, it uses CCA to learn two maximally correlated subspaces, and then learns the logistic regressors in each subspaces. CSR learns a compatible function via structural SVM to get a joint image-text representation and an uniform formulation. GMMFA and GMLDA both use the framework Generalized Multiview Analysis.

4.4 Experimental Setting

In our experiments, we set the parameters for these two tasks as below: $\lambda_1 = \lambda_2 = 3$, $\lambda_3 = 1$, $\lambda_4 = 0.0001$. As for RBF Kernel, we set the RBF bandwidth l_{rbf} as 1 and the variance of noise σ_w^2 as 0 in data preparation for both two model. In addition, we set the dimension of latent variables $c = 200$ for TPLS on Wiki and $c = 300$ for TPLS on Flickr. With regard to the batch setting in SGD, we set $\beta = 100$ for TPLS on Wiki and $\beta = 150$ for TPLS on Flickr.

We use the classical precision-recall curve and the mean average precision (MAP) metric to evaluate the performance of algorithms.

4.5 Result on Wiki

Table 1. The MAP comparison results on Wikipedia database. The results shown in boldface are the best performance.

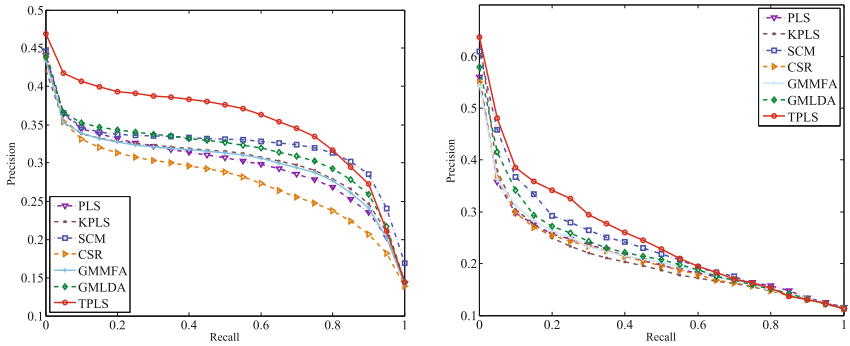
Methods	Tasks		
	im2txt	txt2im	Average
PLS [18]	0.207	0.192	0.199
KPLS [17]	0.260	0.201	0.231
SCM [15]	0.277	0.226	0.252
GMMFA [19]	0.264	0.231	0.248
GMLDA [19]	0.272	0.232	0.253
CSR [23]	0.243	0.201	0.222
TPLS	**0.312**	**0.241**	**0.277**

The MAPs of the different methods on the Wiki dataset are shown in Table 1. From Table 1, we can find the following scenes:

Firstly, the average MAP of KPLS outperform 16.1% than PLS, which indicates that mapping original feature into a high dimensional feature space via kernel function can get better performance. The advantage of KPLS motivates us to construct TPLS based on KPLS rather than PLS.

Secondly, for the supervised methods using the label information, such as SCM, GMMFA and GMLDA, they outperform the unsupervised algorithms PLS and KPLS by at least 7.36%. This indicates that the label information can provide the available information to improve the performance.

Finally, compared with KPLS, TPLS obtained 19.9% higher MAP, which validates the effectiveness of setting the label information as the output variables in TPLS. Compared with the supervised algorithm, the best performance of TPLS outperforms the second best SCM by 12.6% higer MAP in the image-to-text retrieval task. In the text-to-image retrieval task, TPLS outperforms the second best GMLDA by 3.9% higher MAP. All these results show the effectiveness of the constraints in TPLS.



(a) Wiki: PR curves for image query (b) Wiki: PR curves for text query

Fig. 6. Precision recall curves of cross-modal retrieval using both image and text queries on Wiki database.

In Fig. 6(a) and (b), we also show the PR curves of the different methods. We can know that TPLS performs against the other algorithms in both the retrieval tasks at the low levels of the recall. Considering that similar points need to be searched in a small neighborhood of the query, the low levels of recall are more practical in practice.

4.6 Result on Flickr

On the Flickr database, because the high dimension of the image and text features, we use PCA to reduce the dimension of the features. Especially, 95% information energy is preserved by the PCA. In Table 2, we show the MAP of the different methods on the Flickr database. From Table 2, it is easy to find the following scenes:

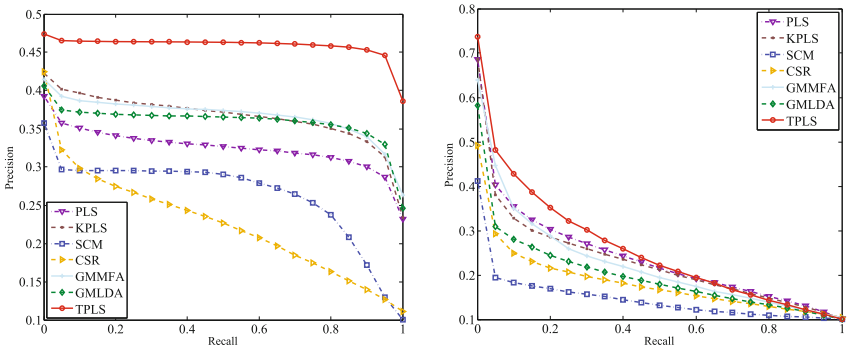
Table 2. The MAP comparison results on Flickr database. The results shown in boldface are the best performance.

Methods	Tasks		
	im2txt	txt2im	Average
PLS [18]	0.269	0.228	0.249
KPLS [17]	0.321	0.219	0.270
SCM [15]	0.215	0.136	0.176
GMMFA [19]	0.311	0.212	0.262
GMLDA [19]	0.299	0.188	0.244
CSR [23]	0.202	0.170	0.186
TPLS	**0.394**	**0.246**	**0.318**

Firstly, KPLS gets at least 3.05% higher MAP than the supervised algorithms, which is different with the conclusion that the supervised methods are often better than the unsupervised methods. In fact, the text features of Flickr is the tag features which also includes the label information. That can further verify the availability of the label information in cross-modal retrieval, which has been indicated by the experiments on Wiki database.

Secondly, besides KPLS, it is remarkable that the average MAP of TPLS is 39.4%, which is 21.4% higher than the second best result (31.1% for GMMFA). This also verifies the effectiveness of our constraints shown by the experiments on Wiki database.

Thirdly, the texts in Filickr database have the tag information, which only uses several words rather than several paragraphs like Wiki databases. Under this condition, TPLS still gains the competitive results on the Flick database, which indicates that TPLS is also effective on the texts which is constructed just by several words.



(a) Flickr: PR curves for image query (b) Flickr: PR curves for text query

Fig. 7. Precision recall curves of cross-modal retrieval using both image and text queries on Flickr database.

Lastly, Fig. 7(a) and (b) show the precision-recall curves of the compared algorithms on two tasks. We can see that with the same recall rate, our approach reach higher precision than other algorithms at low level of recall, which is similar to the results on the Wiki database.

In summary, on both the Wikipedia and Flickr databases, TPLS can achieve the higher performance on both the retrieval tasks, which indicates the effectiveness of TPLS by introducing the label information and reducing the common space difference.

4.7 Running Time Analysis

In the case of the running time for each model, we also carry the experiments to compare it on the Wiki. The results are illustrated in Table 3. Then we can conclude as follow:

Firstly, TPLS solving by SGD is about 20 times faster than that by BGD. It shows the high efficiency of SGD through picking up a small part of training data rather than the whole data set.

Secondly, TPLS is time-consuming in training process. But the training is done offline and only once. Thus the training time cost is not as important as that of the testing time.

Thirdly, TPLS costs similar test time compared with other methods. The contributing factor accounting for the similar test time is that TPLS and other methods all learn a projector and makes test data comparable by multiplying the modal features and the projector directly.

Table 3. Calculational time on the Wiki database. The unit is second.

Methods	Tasks	
	Training time	Testing time
PLS [18]	26.92	17.69
KPLS [17]	84.41	17.76
SCM [15]	12.94	22.77
GMMFA [19]	184.12	16.96
GMLDA [19]	201.92	17.10
TPLS (BGD)	99606.11	16.76
TPLS (SGD)	4297.56	16.59

4.8 Exhibition of Retrieval Result

Besides above experiments, we also show the examples of queries and their results retrieved by RLPLS and GMLDA on the Wiki dataset shown as Fig. 8. The text query and the image of the ground truth are shown in the first column of the first and second row. The top five retrieved images of GMLDA and RLPLS are exhibited at the first and second row, respectively. The images with red frames are the wrong retrieval results based on their respective class. From the figure, we can know that all the retrieved images of RLPLS are correct while only the third retrieved image of GMLDA is related to the text query. At the third and forth row, we also show the image query, the text of the ground truth and the top five retrieved documents shown with their corresponding images of RLPLS and GMLDA. From the figure, we can also find that the third images of RLPLS is a wrong retrieval result.

Fig. 8. Two examples of queries and their results retrieved by RLPLS on the Wiki dataset. (Color figure online)

5 Conclusion

In this paper, we proposed a novel method to solve cross-modal retrieval problem under the hardware condition of the mobile devices, and apply it to the image-text cross-modal retrieval. In our approach, we regard the label information as the third modality so as to construct three KPLS between any two modalities. Furthermore, we add the distance constrain of target space so as to achieve learning the consistent feature representation in cross-modal retrieval. Experiments are carried out on two databases, Wikipedia and Flickr, showing that our proposed algorithm performs against existing competitive algorithms.

Later, we will look for more effective learning model to learn consistent representation. And we will design model to solve multi-label and cross-modal retrieval in mobile device.

Acknowledgement. This work was supported in part by the National Natural Science Foundation of China under Grant 61672497, Grant 61332016, Grant 61620106009, Grant 61650202 and Grant U1636214, in part by the National Basic Research Program of China (973 Program) under Grant 2015CB351802, and in part by the Key Research Program of Frontier Sciences of CAS under Grant QYZDJ-SSW-SYS013. This work was also partially supported by CAS Pioneer Hundred Talents Program by Dr. Qiang Qu.

References

1. Bai, S., Bai, X.: Sparse contextual activation for efficient visual re-ranking. IEEE Trans. Image Process. **25**(3), 1056–1069 (2016)
2. Bai, X., Bai, S., Zhu, Z., Latecki, L.: 3d shape matching via two layer coding. IEEE Trans. Pattern Anal. Mach. Intell. **37**(12), 2361–2373 (2015)
3. Chen, Y., Wang, L., Wang, W., Zhang, Z.: Continuum regression for cross-modal multimedia retrieval (ICIP 2012), pp. 1949–1952 (2012)

4. Deng, J., Du, L., Shen, Y.: Heterogeneous metric learning for cross-modal multimedia retrieval. In: International Conference on Web Information Systems Engineering, pp. 43–56 (2013)
5. Duan, L., Xu, D., Tsang, I.: Learning with augmented features for heterogeneous domain adaptation. arXiv preprint arXiv:1206.4660 (2012)
6. Gong, Y., Lazebnik, S.: Iterative quantization: a procrustean approach to learning binary codes. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 817–824 (2011)
7. Hardoon, D.R., Szedmak, S., Shawe-Taylor, J.: Canonical correlation analysis: an overview with application to learning methods. Neural Comput. **16**(12), 2639–2664 (2004)
8. He, R., Zhang, M., Wang, L., Ye, J., Yin, Q.: Cross-modal subspace learning via pairwise constraints. IEEE Trans. Image Process. **24**(12), 5543–5556 (2015). A Publication of the IEEE Signal Processing Society
9. Jia, Y., Salzmann, M., Darrell, T.: Learning cross-modality similarity for multinomial data. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 2407–2414 (2011)
10. Kang, C., Xiang, S., Liao, S., Xu, C., Pan, C.: Learning consistent feature representation for cross-modal multimedia retrieval. IEEE Trans. Multimed. **17**(3), 370–381 (2015)
11. Li, A., Shan, S., Chen, X., Gao, W.: Cross-pose face recognition based on partial least squares. Pattern Recognit. Lett. **32**(15), 1948–1955 (2011)
12. Lu, X., Wu, F., Tang, S., Zhang, Z., He, X., Zhuang, Y.: A low rank structural large margin method for cross-modal ranking, pp. 433–442 (2013)
13. Mao, X., Lin, B., Cai, D., He, X., Pei, J.: Parallel field alignment for cross media retrieval. In: Proceedings of the ACM International Conference on Multimedia, pp. 897–906 (2013)
14. Pereira, J.C., Coviello, E., Doyle, G., Rasiwasia, N., Lanckriet, G., Levy, R., Vasconcelos, N.: On the role of correlation and abstraction in cross-modal multimedia retrieval. IEEE Trans. Pattern Anal. Mach. Intell. **36**(3), 521–535 (2014)
15. Rasiwasia, N., Pereira, J.C., Coviello, E., Doyle, G., Lanckriet, G.R.G., Levy, R., Vasconcelos, N.: A new approach to cross-modal multimedia retrieval. In: Proceedings of the ACM International Conference on Multimedia, pp. 251–260 (2010)
16. Rosipal, R., Krämer, N.: Overview and recent advances in partial least squares. In: Subspace, Latent Structure and Feature Selection, pp. 34–51 (2006)
17. Rosipal, R., Trejo, L.J.: Kernel partial least squares regression in reproducing kernel hilbert space. J. Mach. Learn. Res. **2**, 97–123 (2002)
18. Sharma, A., Jacobs, D.W.: Bypassing synthesis: PLS for face recognition with pose, low-resolution and sketch. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 593–600 (2011)
19. Sharma, A., Kumar, A., Daume III, H., Jacobs, D.W.: Generalized multiview analysis: a discriminative latent space. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2160–2167. IEEE (2012)
20. Song, G., Wang, S., Huang, Q., Tian, Q.: Similarity gaussian process latent variable model for multi-modal data analysis. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 4050–4058 (2015)
21. Tang, J., Wang, H., Yan, Y.: Learning hough regression models via bridge partial least squares for object detection. Neurocomputing **152**, 236–249 (2015)
22. Tenenbaum, J.B., Freeman, W.T.: Separating style and content with bilinear models. Neural Comput. **12**(6), 1247–1283 (2000)

23. Verma, Y., Jawahar, C.: Im2text and text2im: associating images and texts for cross-modal retrieval. In: Proceedings of the British Machine Vision Conference (2014)
24. Viresh, R., Nikhil, R., Jawahar, C.V.: Multi-label cross-modal retrieval. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 4094–4102 (2015)
25. Wang, J., Kumar, S., Chang, S.: Semi-supervised hashing for large-scale search. IEEE Trans. Pattern Anal. Mach. Intell. **34**(12), 2393–2406 (2012)
26. Wang, K., He, R., Wang, W., Wang, L., Tan, T.: Learning coupled feature spaces for cross-modal matching. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 2088–2095 (2013)
27. Wang, S., Zhuang, F., Jiang, S., Huang, Q., Tian, Q.: Cluster-sensitive structured correlation analysis for web cross-modal retrieval. Neurocomputing **168**, 747–760 (2015)
28. Xie, L., Pan, P., Lu, Y.: A semantic model for cross-modal and multi-modal retrieval. In: Proceedings of the ACM Conference on International Conference on Multimedia Retrieval, pp. 175–182 (2013)
29. Yao, T., Kong, X., Fu, H., Tian, Q.: Semantic consistency hashing for cross-modal retrieval. Neurocomputing **193**, 250–259 (2016)
30. Yu, Z., Zhang, Y., Tang, S., Yang, Y., Tian, Q., Luo, J.: Cross-media hashing with kernel regression. In: IEEE International Conference on Multimedia and Expo, pp. 1–6 (2014)
31. Zhang, H., Liu, Y., Ma, Z.: Fusing inherent and external knowledge with nonlinear learning for cross-media retrieval. Neurocomputing **119**, 10–16 (2013)
32. Zhang, L., Ma, B., He, J., Li, G., Huang, Q., Tian, Q.: Adaptively unified semi-supervised learning for cross-modal retrieval. In: International Conference on Artificial Intelligence, pp. 3406–3412 (2017)
33. Zhang, L., Ma, B., Li, G., Huang, Q., Tian, Q.: Pl-ranking: a novel ranking method for cross-modal retrieval. In: Proceedings of the ACM International Conference on Multimedia, pp. 1355–1364 (2016)
34. Zhang, L., Ma, B., Li, G., Huang, Q., Tian, Q.: Cross-modal retrieval using multi-ordered discriminative structured subspace learning. IEEE Trans. Multimed. **19**(6), 1220–1233 (2017)
35. Zhuang, Y., Wang, Y., Wu, F., Zhang, Y., Lu, W.: Supervised coupled dictionary learning with group structures for multi-modal retrieval. In: AAAI Conference on Artificial Intelligence (2013)