Image Retrieval and Ranking via Consistently Reconstructing Multi-attribute Queries

Xiaochun Cao[1,2], Hua Zhang[1,⋆], Xiaojie Guo[2], Si Liu[3], and Xiaowu Chen[4]

[1] School of Computer Science and Technology, Tianjin University, Tianjin, China
[2] State Key Laboratory of Information Security, IIE, Chinese Academy of Sciences, China
[3] Department of Electrical & Computer Engineering, National University of Singapore
[4] State Key Laboratory of Virtual Reality Technology and Systems School of Computer Science and Engineering, Beihang University, Beijing, China
caoxiaochun@iie.ac.cn, huazhang@tju.edu.cn, xj.max.guo@gmail.com, dcslius@nus.edu.sg, chen@buaa.edu.cn

Abstract. Image retrieval and ranking based on the multi-attribute queries is beneficial to various real world applications. Traditional methods on this problem often utilize intermediate representations generated by attribute classifiers to describe the images, and then the images in the database are sorted according to their similarities to the query. However, such a scheme has two main challenges: 1) how to exploit the correlation between query attributes and non-query attributes, and 2) how to handle noisy representations since the pre-defined attribute classifiers are probably unreliable. To overcome these challenges, we discover the correlation among attributes via expanding the query representation, and imposing the group sparsity on representations to reduce the disturbance of noisy data. Specifically, given a multi-attribute query matrix with each row corresponding to a query attribute and each column the pre-defined attribute, we firstly expand the query based on the correlation of the attributes learned from the training data. Then, the expanded query matrix is reconstructed by the images in the dataset with the $\ell_{2,1}$ regularization. Furthermore, we introduce the ranking SVM into the objective function to guarantee the ranking consistency. Finally, we adopt a graph regularization to preserve the local visual similarity among images. Extensive experiments on LFW, CUB-200-2011, and Shoes datasets are conducted to demonstrate the effectiveness of our proposed method.

Keywords: Multi-Attribute Image, Image Retrieval & Ranking, Group Sparsity.

1 Introduction

The goal of image retrieval based on multi-attribute queries is to, from a database, recall images semantically similar to the query in a ranked order. It is beneficial yet challenging to many computer vision applications [8,17,15,14,23]. Different from single-attribute queries, a multi-attribute query can exploit the correlation among the query attributes and preferentially recommend the images similar to the whole attribute query. A traditional framework [16,9] first trains several attribute classifiers to describe the

⋆ Corresponding author.

D. Fleet et al. (Eds.): ECCV 2014, Part I, LNCS 8689, pp. 569–583, 2014.

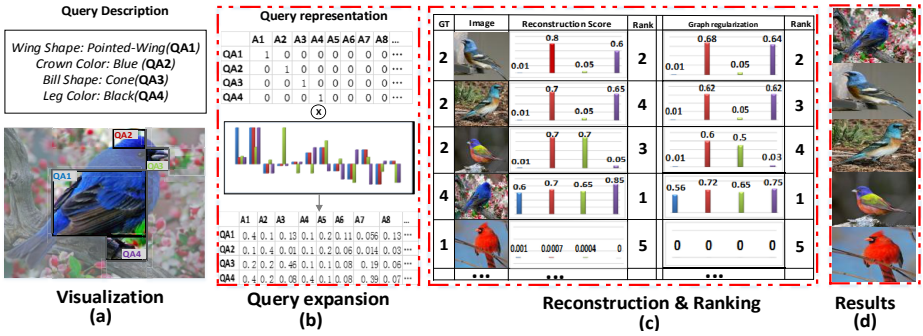


Fig. 1. An illustration of our proposed framework. Given a multi-attribute query (a), we first describe the query by a matrix. And then the query is expanded based on the correlation among attributes (Sec. 2.1). After that, we reconstruct the query by incorporating the ranking regularization into the objective function (Sec. 2.2 & 2.3). Furthermore, the graph regularization is introduced to enforce the local similarity (Sec. 2.4). Each column of the matrix (subgraph (b), top) indicates a visual attributes (A_j), while each row a query attribute (QA_i). In subgraph (c), GT is the number of query attributes in the corresponding images. The reconstruction score denotes the reconstruction values with ranking regularization. Graph regularization represents the reconstruction values with graph regularization. Finally, the retrieval results are shown in subgraph (d).

images, based on which, the candidate images are orderly feedback according to their similarities to the query. To improve the performance of retrieval, [17,23] introduce the non-query attributes as the context information to complement the query ones. More recently, [15] focuses on constructing more discriminative query classifiers by considering the distance among the query attributes. Impressive results have been achieved by previous work [17,15,14,23,16,9]. However, there are still three factors needed to be considered, which would significantly influence the retrieval and ranking performance. Firstly, due to the influence of various viewpoints, scales and occlusions, the trained classifier might not be sufficiently accurate. Secondly, considering the attributes in the query independently might lose the structure information. Thirdly, the correlation between the query and non-query attributes contains not only the co-occurrence but also the mutual inference correlation. In other words, a query attribute could be inferred by other attributes. For example, "Female" could be collated with "No Beard", "Wearing Lipstick", and "Wearing Necklace" *etc*.

In this paper, we propose a novel framework to handle imprecise and noisy image representations, consider the dependence among query attributes, and explicitly exploit the correlation between the query and non-query attributes. As shown in Fig. 1, our system has four main components: *Query expansion*, *Query reconstruction*, *Rank regularization*, and *Graph regularization*. The input is the multi-attribute query (QA_1,...,QA_4) as shown in Fig. 1 (a). In the first component, we first represent them as a matrix (top, Fig. 1(b)), each row of which is a query attribute (QA_i), and each column denotes a pre-defined attribute (A_j). Motivated by [22,11,4] on tag completion, we propose an attribute query expansion technique to expand the original query by using the correlation (middle, Fig. 1(b)). In the second component, we compute the similarity between the

expanded query and the images in the database by using $\ell_{2,1}$ norm based reconstruction framework. This is inspired by the success of [19] in image annotation community. [19] shows that the semantic similarity between two images with overlapped labels can be well recovered in a reconstruction way. In the third component, since our goal is to retrieve and rank the similar images, beside the reconstruction error, the ranking error also needs to be considered in the objective function. We utilize the inequally contribution of different query reconstruction coefficients to boost the performance similar to [17,23]. In the fourth component, we introduce a graph regularization based on the low-level features of images into the objective function. It enforces the similar images to have similar reconstruction coefficients and rankings. The reason of introducing graph regularization is that two visually similar images might generate different intermediate representations because of various factors such as occlusions, scales. Extensive experiments are designed to validate the superior performance of our method, compared to state of the art alternatives, on LFW [10], CUB-200-2011 Bird [20], and Shoes [1] datasets.

The contributions of our work can be summarized as follows:

- We develop a reconstruction based framework to retrieve images by multi-attribute queries;
- A $\ell_{2,1}$ based constraint is introduced to retain the semantic structure of query;
- The query of our framework can be either the multi-attribute or the image.

1.1 Related Work

In this part, we briefly review the previous work that are closely related to ours. These work can be roughly divided into three groups:

1) *Attribute Classifier Based Image Retrieval*. In recent years, numerous algorithms in this category have been proposed, which can be further grouped into two categories. One focuses on modeling the multi-attribute query. Scheirer *et al.* [16] propose an attribute score calibration approach by considering the distribution of different detectors of the attribute. This method models the distribution of the query by taking advantage of the opposite side of the query attribute. However it is nontrivial to find the opposite of each attribute. [15] constructs the discriminative attribute detectors by analyzing the feature distance between attributes. As this method depends on the attribute distance, its performance is sensitive to the amount of training data. The other category is to explore the correlation among attributes: To model such interdependencies, Siddiquie *et al.* [17] design an image ranking and retrieval method by taking into account both query attributes and non-query attributes, which exploits the positive and negative correlations between the query and non-query attributes to improve the discriminative of image representation. Further, [23] introduces weak attributes to describe images by class names. This approach can handle large scale images and automatically select the correlation features. The approaches proposed in [17,23] focus on the presence of selected query attributes without the consideration of that the consistency of similar image would generate the similarity rank.

2) *Reconstruction Based Image Annotation and Retrieval*. Wang *et al.* [19] propose to annotate the query image by using sparse coding framework via the ℓ_1 norm. The

differences between our work are that first we want to retrieve the images based on the attribute query. Second, we use the group sparsity by $\ell_{2,1}$ norm instead of ℓ_1 norm. Zhang *et al.* [25] propose a group sparsity based method for the image annotation and retrieval task. In [25], features are divided into several groups to find the discriminative representation group by the constructed image pairs. In the testing phase, it automatically computes the similarity based on the feature weight between the test image and other images. Comparing with [25], our method uses not only the reconstruction error but also the ranking error and graph regularization into the objective function.

3) *Tag completion*. In [22], the authors propose to firstly complete the image tags based on the co-occurrence among tags, and then compute the similarity between the query and the images by using the tag relevance. The difference is that [22] uses the manual tags while our method utilizes the attribute. Moreover, [11] proposes to complete the image tags based on the image-tag association and tag-tag concurrence. And [4] gives an efficient approach by introducing a co-regularized framework. The difference between [11,4] and our method is that they focus on image annotation instead of image retrieval which would lead to the different framework.

2 Our Method

A multi-attribute query is represented by a matrix $\mathbf{Q} \in \{0,1\}^{q\times m}$, where q is the number of query attributes and m denotes the feature dimension. The element $Q_{ij} = 1$ when the pre-defined attribute (A_j) is selected as the i^{th} query attribute (QA_i). For example, in the Bird dataset, m is the number of pre-defined attributes, *i.e.* 312. If the query is "a bird with pointed-wing shape, blue crown, cone bill shape, and black leg" as shown in Fig. 1, we extract four attributes: "Wing shape: Pointed-Wing", "Crown Color: Blue", "Bill Shape: Cone", and "Leg color: Black", *i.e.* q is 4. Furthermore, Let $\mathbf{D} \in \mathbb{R}^{m\times N}$ represent a set of N training images, each of which is described by m-dimensional attribute scores.

2.1 Query Expansion

To characterize the correlation between query attributes and non-query attributes, we firstly collect the positive examples for the query from the training data $\mathbf{D}$, denoted as $\mathbf{D_q} \in \mathbb{R}^{m\times N_q}$, where N_q is the number of positive images. An image is positive when it contains all the query attributes. The goal is to compute the query dependent correlation matrix $\mathbf{B} \in \mathbb{R}^{m\times m}$. This problem is formalized as reconstructing $\mathbf{D_q} \in \mathbb{R}^{m\times N_q}$ from a "corrupted" $\widetilde{\mathbf{D}}_q \in \mathbb{R}^{m\times N_q}$. In order to mine the correlations among the attributes, we empty all the rows in $\mathbf{D}_q$ which represent the query attributes QA_i to obtain $\widetilde{\mathbf{D}}_q$ as:

$$\widetilde{\mathbf{D}}_q = (\mathbf{1}_{N_q\times m} - \mathbf{1}_{N_q\times q}\mathbf{Q})^T \odot \mathbf{D_q}, \tag{1}$$

where $\mathbf{1}_{r\times c}$ denotes a (r, c) matrix full of ones and $\odot$ is the Hadamard product. The correlation matrix $\mathbf{B}$ is expected to catch the correlation between non-query attributes in $\widetilde{\mathbf{D}}_q$ and query attributes in $\mathbf{D}$:

$$\mathbf{B} = \operatorname{argmin} \|\mathbf{D}_q - \mathbf{B}\widetilde{\mathbf{D}}_q\|^2 + \lambda\|\mathbf{B}\|_2, \tag{2}$$

where the second term is used to prevent overfitting and λ is the weighting parameter. This is an ordinary least squares regressor which has the closed form solution $\mathbf{B} = (\widetilde{\mathbf{D}}_q^T\widetilde{\mathbf{D}}_q + \lambda\mathbf{I})^{-1}\widetilde{\mathbf{D}}_q^T\mathbf{D_q}$. Finally, we obtain the expanded query matrix $\mathbf{BQ}^T$ as shown in Fig. 1 (b) bottom.

2.2 Query Reconstruction Based on Inducing Group Sparsity

Computing the similarity measure between the images and the expanded query by directly using the predicted attribute vectors might not handle the imprecise and noisy image representations caused by *e.g.* scale, occlusion. To overcome this limitation, we adopt a sparse reconstruction framework which has been proven to be effective in [19,21]. We use training data $\mathbf{D}$ as the base to sparsely reconstruct the expanded query attributes $\mathbf{BQ}^T$ with $\ell_{2,1}$ norm constraint:

$$\mathbf{X} = \text{argmin}\,\|\mathbf{BQ^T} - \mathbf{DX}\|_2^2 + \alpha\|\mathbf{X}\|_{2,1}, \tag{3}$$

where $\mathbf{X} \in \mathbb{R}^{N\times q}$ is the reconstruction coefficient, $\|\cdot\|_{2,1}$ is the $\ell_{2,1}$ norm, and α is the weighting parameter. The training images which have non-zero reconstruction coefficients are considered semantically related to the query. Specially, each row of $\mathbf{X}$ refers to the similarity scores of images with respect to corresponding query attributes.

In our work, we use $\ell_{2,1}$ norm instead of ℓ_1 for sparse recovery. The reasons of selecting $\ell_{2,1}$ norm regularization are twofold. Firstly, each image (column of $\mathbf{D}$) is restricted to be evenly similar to the expanded query attributes as imposed by the ℓ_2 norm. Secondly, the ℓ_1 norm is used to sum the similarities across images, and select only the reliable images by removing imprecise and noisy image representations. The group structure generated by $\ell_{2,1}$ norm ensures more robust and accurate results.

2.3 Ranking Regularization

Now, we have the reconstruction coefficient $\mathbf{X}$ whose non-zero rows indicate the images related to the query, moreover the value of each element denotes the degree of similarity. To rank the retrieved images based on their similarity to the query, an intuitive way is to sum the coefficient values and sort them. The assumption behind this way is that all the reconstruction coefficients contribute equally to the rank. We introduce a weighting scheme by ranking SVM [7] which embeds the ranking in its loss function. The training images are firstly collected from the training part of each dataset, which have the groundtruth labels. Then we define each query $\mathbf{Q}_t \subset Q$, where Q is the set of queries. The objective function of ranking SVM is:

$$\text{argmin}\ \ \mathbf{W}^T\mathbf{W} + C\sum_t \xi_t,\ \ \forall t\ \ \mathbf{W}^T(\phi(\mathbf{Q}_t, \mathbf{y}_t^\star) - \phi(\mathbf{Q}_t, \mathbf{y}_t)) \geq \Delta(\mathbf{y}_t^\star, \mathbf{y}_t) - \xi_t, \tag{4}$$

where $\mathbf{W} \in \mathbb{R}^{q\times q}$ is the ranking weight, $\mathbf{y}^\star$ is the set of images which contains all the constituent attributes in query $\mathbf{Q}_t$. $\mathbf{y}$ represents the other set of images which do not include all the attributes in query $\mathbf{Q}_t$. $\Delta(\mathbf{y}_t^\star, \mathbf{y}_t)$ is the loss function as in [17,14]. C is a parameter which determines the trade-off between training accuracy and regularization.

ξ_t is a slack variable to handle the soft margin. $\phi(\cdot)$ is the feature map of images. Eq. 4 could be efficiently solved by cutting plane method [18].

Specifically, in our method we use the reconstruction coefficient $\mathbf{X}$ as the feature map. Then we can use the loss function to penalize outputs $\mathbf{y}_t$ that deviate from the correct output $\mathbf{y}_t^\star$ based on the performance metric we want to optimize for. We set the loss function as hamming loss:

$$\Delta(\mathbf{y}_t^\star, \mathbf{y}_t) = 1 - \frac{|\mathbf{y}_t \cap \mathbf{y}_t^\star| + |\bar{\mathbf{y}}_t \cap \bar{\mathbf{y}}_t^\star|}{N}. \tag{5}$$

The reason to choose hamming loss is that it computed efficiently which only needs $O(|y_l|)$ to solve Eq. 5 as discussed in [17,23], where $|y_l|$ is the number of training images. Eq. 3 only considers the reconstruction error while our aim is to rank the images. Motivated by [6] which merged the classification error to objective function to achieve a better classification results, we also embed the ranking error to the objective function which is denoted as $\Omega(\mathbf{W}, \mathbf{X})$. We define $\mathbf{y}_s$ as the prediction results of after computing the scores of training images $\mathbf{XW}$, and $\mathbf{y}_{gt}$ as the groundtruth ranking of training images under the current query $\mathbf{Q}_t$. The Normalized Discounted Cumulative Gains [3] is adopted as the ranking metric:

$$NDCG@k = \sum_{j=1}^{k} \frac{2^{rel(j)} - 1}{log(1 + j)}, \tag{6}$$

where $rel(j)$ is the relevance of the j^{th} ranked image and Z is a normalization constant. And $NDCG@k$ represents the score of top k ranked images. The relevance is defined as the number of shared attributes between the query and the images. Then, $\Omega(\mathbf{W}, \mathbf{X}) = 1 - NDCG@k(\mathbf{y}_s, \mathbf{y}_{gt})$ where $k = 100$ in all our experiments.

To incorporate the ranking loss into the objective function, we could set $\Omega(\mathbf{W}, \mathbf{X})$ as a regularization term. Then the objective function Eq. 3 is rewritten as:

$$(\mathbf{W}, \mathbf{X}) = \text{argmin} \, \|\mathbf{BQ}^\mathbf{T} - \mathbf{DX}\|_2^2 + \alpha\|\mathbf{X}\|_{2,1} + \gamma\Omega(\mathbf{W}, \mathbf{X}), \tag{7}$$

where γ is the weighting parameter.

2.4 Graph Regularization

Since we use the score of attribute classifiers as the intermediate representation ($\mathbf{D}$) of the image, one limitation is that the visually similar images might generate different representations because of various factors such as occlusions and scales. Preserving the local visual similarity among images would boost the performance of our method. That is to say, when the images are visually similar, they should have similar scores to the query. Moreover, in the test phase, we utilize the such locality to make our method insensitive to the data variance. Graph regularization [26] is designed for such purpose in the reconstruction step. In our work, the goal of introducing the graph regularization into the objective function is to ensure the visually similar images to obtain the close rankings.

Mathematically, each image $\mathbf{f}_i = [h_1, h_2, ..., h_s]^T \in \mathbb{R}^{s\times 1}$ is represented by its low-level features, such as HOG, SIFT and Texton, where s denotes the dimension of the low level feature representation. We construct a graph $\mathbf{G}$ based on the nearest neighborhood method using the low-level feature distance, each vertex of which stands for an image. Let $\mathbf{S}$ be the weight matrix of graph $\mathbf{G}$, where $S_{i,j}$ is set to 1 when $\mathbf{f}_i$ is one of the k-nearest neighbors of $\mathbf{f}_j$ or equivalently $\mathbf{f}_j$ is one of the k-nearest neighbors of $\mathbf{f}_i$, otherwise $S_{i,j} = 0$. We define the degree of the image as $\mathbf{J} = diag\{j_1, j_2, ..., j_N\}$, where $j_n = \sum_{i=1}^{N} S_{ij}$. By considering the goal of this term, a reasonable method for choosing a graph regularizer is to minimize the following objective function:

$$\Psi(\mathbf{W}, \mathbf{X}) = \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}((\mathbf{x}_i - \mathbf{x}_j)^2 S_{ij} + (\mathbf{x}_i\mathbf{W} - \mathbf{x}_j\mathbf{W})^2 S_{ij}) = Tr((\mathbf{XW})^{\mathbf{T}}\mathbf{L}(\mathbf{XW}) + \mathbf{X}^{\mathbf{T}}\mathbf{LX}), \tag{8}$$

where $\mathbf{L} = \mathbf{S} - \mathbf{J}$ is the Laplacian matrix. $\mathbf{x}_i \in \mathbb{R}^{1\times q}$ denotes the reconstruction coefficient with respect to the i^{th} image. This regularization enforces the similar images to obtain the close reconstruction coefficients and ranking scores. By incorporating the Laplacian regularizer Ψ into the objective (7), we have the final objective function:

$$(\mathbf{X}, \mathbf{W}) = \text{argmin}\underbrace{||\mathbf{BQ}^{\mathbf{T}} - \mathbf{DX}||_2^2 + \alpha||\mathbf{X}||_{2,1}}_{sparse\ reconstruction} + \underbrace{\gamma\Omega(\mathbf{W}, \mathbf{X})}_{ranking\ regularization} + \underbrace{\beta\Psi(\mathbf{W}, \mathbf{X})}_{graph\ regularization}. \tag{9}$$

Recall that $\Omega(\mathbf{W}, \mathbf{X})$ denotes the ranking regularization term constructed based on the standard ranking SVM [7]. $\Psi(\mathbf{W}, \mathbf{X})$ is defined as the graph regularizer, which is used to keep the locality of the images. $\lambda, \alpha, \beta, \gamma$ are the weights associated with each regularization term.

2.5 Optimization

Since the local graph and global ranking consistency constraints in the objective function are quadratic, the objective function Eq. 9 is a convex problem, thus it could be solved by a gradient descent-based approach as shown in Algorithm 1.

Firstly, Eq. 2 could be considered as a rigid regression problem, which could be efficiently solved by gradient descent to get the correlation matrix $\mathbf{B}$. Then Eq. 9 is composed of two variables $(\mathbf{W}, \mathbf{X})$ to be optimized. We use an iterative strategy to solve it:

-Holding $\mathbf{W}$ fixed, learn the reconstruction coefficient $\mathbf{X}$ by solving a standard sparse coding problem. We use a similar algorithm described in [12] to solve this optimize problem.

-Holding $\mathbf{X}$ fixed, learn the ranking weight parameter $\mathbf{W}$ by employing a standard ranking SVM, which was proposed in [7].

The two steps are repeated until convergence. Furthermore, we observe that there needs to be about $2 \sim 4$ iterations to convergence.

2.6 Test Phase

In the test phase, we firstly extract the attribute presentation $\mathbf{D}_t$ by using attribute classifiers. Then the similarity among testing images is computed based their low-level

feature representation and therefore we have $\mathbf{S}_t$ and $\mathbf{L}_t$. Since we obtained the ranking weighting coefficients $\mathbf{W}$, query representation $\mathbf{BQ^T}$ and the Laplacian matrix $\mathbf{L_t}$, we just need to solve a sparse reconstruction problem:

$$\mathbf{X}_t = \text{argmin}\,||\mathbf{BQ^T} - \mathbf{D_tX_t}||_2^2 + \alpha||\mathbf{X_t}||_{2,1} + \beta Tr((\mathbf{X_tW})^\mathbf{T}\mathbf{L_t}(\mathbf{X_tW}) + \mathbf{X_t^T}\mathbf{L_tX_t}), \quad (10)$$

where $\mathbf{X}_t$ represents the reconstruction coefficients for the testing images. Finally, we use the reconstruction coefficients $\mathbf{X_t}$ and the ranking weight $\mathbf{W}$ to get the final image ranking list. α and β are determined on the validation dataset.

Algorithm 1. The main training steps of our method

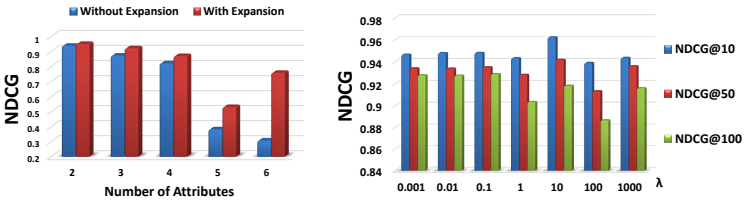
1. **Input**: $\mathbf{B} \in \mathbb{R}^{m\times m}, \mathbf{Q} \in \mathbb{R}^{q\times m}, \mathbf{D} \in \mathbb{R}^{m\times n}, \lambda, \alpha, \beta, \gamma$
2. Initialize $\mathbf{W} = \mathbf{1}_{q\times q}, \mathbf{X} = \mathbf{0}$
3. **repeat**
4. Computing reconstruction coefficients $\mathbf{X}$ according to Eq. 9 with $\mathbf{W}$ fixed.
5. Computing image ranking weight $\mathbf{W}$ according to Eq. 4 with $\mathbf{X}$ fixed.
6. Update the solutions $\mathbf{W}$ and $\mathbf{X}$.
7. **until** convergence:$||\mathbf{BQ^T} - \mathbf{DX}||_2 \leq \epsilon$ & $\Omega(\mathbf{W}, \mathbf{X}) \leq \rho$
8. **Output**:Reconstruction coefficients $\mathbf{X}$, Ranking weight $\mathbf{W}$;

3 Experiments

We have conducted extensive experiments to evaluate our image retrieval and ranking framework on three public datasets LFW [10], CUB-200-2011 [2], and Shoes dataset [1]. First, we show the effect of parameters to the performance. Second, compared with the state-of-the-art approaches, we validate the superior performance of our method. Third, we experiment on the shoes dataset to demonstrate our method can be applied on the case directly using images as queries.

3.1 Experimental Settings

Dataset. LFW[10] (Labeled Faces in the Wild) is originally constructed for face verification. In our experiments, a subset of images consisting of 9992 images is selected. In addition, each image is annotated with 73 attributes. We simply divide this subset into two parts: 50% of all the images are randomly chosen as the training data and the remaining are used for testing. **CUB-200-2011** [2] (Caltech-UCSD Birds-200-2011) is composed of 11,788 images from 200 bird categories, which are the uncropped images of birds with various statuses, such as flying, perched, swimming, truncated and occluded, in the wild. Furthermore, each bird has been described by 312 binary visual attributes. **Shoes dataset** [1]: contains 14,765 images of 10 classes of shoes. The images in this dataset are relatively clean without confounding visual challenges like clutters, occlusions *etc*.



(a) (b)

Fig. 2. Evaluation of query expansion parameter: (a) The comparison results between with query expansion (Red histogram) and without query expansion (Blue histogram). The comparison is constructed on five types of queries. (b) NDCG Varies for Different Parameters: The weight of regularization λ and the number of top images.

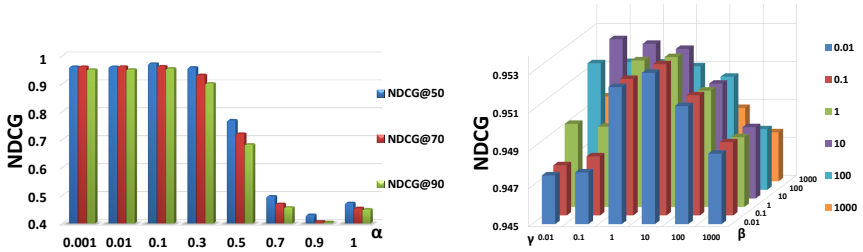
Query. Since there are no pre-defined multi-attribute queries available for our task, we create the queries by ourselves based on the training part of each dataset. Two kinds of queries, *i.e.* semantic attribute queries (on LFW and CUB-200-2011) and image queries (on Shoes) are considered. For the semantic attribute ones, 5 different structures are involved, which contain various numbers, from 2 to 6, of attributes in the query. Specially, we randomly select the attributes from the training set to construct the query with the constraint that each query should contain at least 30 related images. For the image query, we randomly select the images from the training set used as queries.

Evaluation Metric: Instead of using a binary relevance, we design the relevance with multiple levels [13]. The more attributes an image shares with the query, the heavier the relevance is. As the common attributes become less, the relevance decreases. By considering that our goal is to rank the image, we select Normalized Discounted Cumulative Gains (NDCG) [3] as the evaluation criterion.

3.2 Component Evaluation

In this part, we use the validation dataset for tuning parameters and analyzing their corresponding influences to image retrieval and ranking performance. There are mainly four parameters of our proposed model including λ, α, β, and γ. λ controls the weight to avoid over-fitting when we expand query. α corresponds the sparsity of the representation, γ takes care of the contribution of ranking error in the objective function , while β determines the importance of the locality similarity among images. To evaluate the influence of each parameter, we randomly select 100 queries which are composed of two attributes from LFW. As for C in Eq. 4 and the number of nearest neighborhoods k in the graph regularization, we empirically set them to 0.1 and 5, respectively.

Query Expansion Evaluation. Firstly, we conduct experiments in this part to validate the advantage of the usage of the query expansion. In this validation, we test queries from 5 kinds of query structures, each of which consists of 100 queries. The experiment results are shown in Fig. 2(a). The advantage of the query expansion is not distinct when the number of attributes is small. However, as the number of query attributes grows, the query expansion shows its power. The reason is that when the query attributes are few,



(a) (b)

Fig. 3. Evaluation of weighting parameters in the objective function. (a) NDCG Varies for Different Sparsity: The higher value of α indicates more sparsity. (b) NDCG@100 results of different combinations of parameters β and γ.

for instance only two attributes, there are a lot of related images that limits the improvement. However, once the attributes becomes more, the number of the related images sharply drops, then the advantage of expansion gets outstanding. Besides, there is one more important parameter λ directly affecting the performance of the query expansion. The selected value of λ ranges from 10^{-3} to 10^{3} as shown in Fig. 2(b). The performance indicates that the query expansion is easily over-fitted when the value of λ is small.

Query Reconstruction Evaluation. We further investigate the influence of the group sparsity by varying α. Smaller value of α indicates that more images are involved in the ranking process, which would introduce more noises and increase the training time. While larger value of α would make a smaller subset of images response, which would give a desired ranking performance. As shown in Fig. 3(a), the optimal setting for α is around 0.1. Based on the results, the conclusion that the sparsity could help for preserving the related images and keeping the structure discriminative can be drawn. We also show how our algorithm performs given different β and γ values. The ranges for the two parameters are all set to be from 0.01 to 1000. From the results shown in Fig. 3(a), we observe that our algorithm achieves the best results when $\beta = 10$ and $\gamma = 0.1$. Please note that the bins in different colors in Fig. 3(a) represent different values of γ.

3.3 LFW Dataset

In this section, we test the performance of our method on LFW. The images in this dataset are evenly separated into two parts including the training set and the testing. We adopt the attributes defined in [10], say 73-dimensional scores of the attribute classifiers. We randomly generate 2,000 queries to do the experiment. The number of queries from two to six attributes are 500, 400, 500, 300 and 300, respectively. The basic parameters of our model are set as $\lambda = 10$ the query expansion process, and $\alpha = 0.1, \beta = 10.5, \gamma = 0.5$ in the training. These parameters are fixed throughout this experiment.

Three related work on multi-attribute based image retrieval and ranking, *i.e. RMLL* [14], *MARR* [17] and *Weak Attributes* [23] are compared with our method. The parameters of the competitors are from the corresponding papers. To reveal the advantages of fusing the ranking regularization and graph regularization into the objective function,

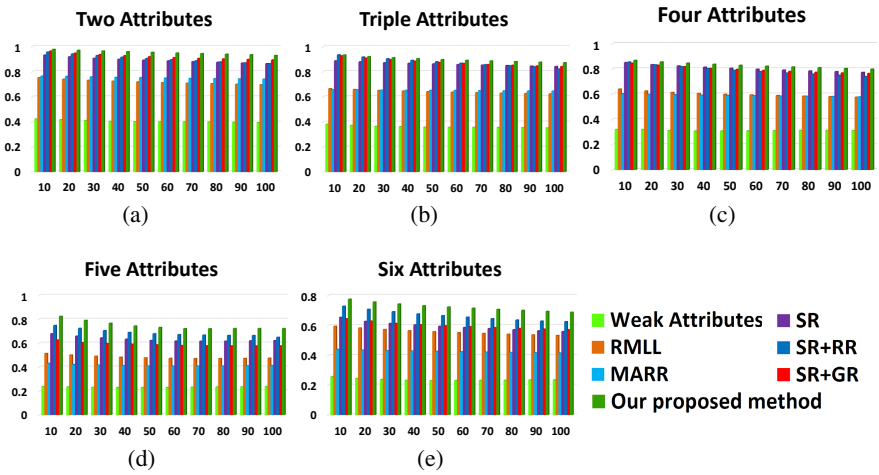


Fig. 4. The comparison results between our proposed methods and the state-of-the-art on LFW dataset. From (a)-(e) are the results of five types of queries. Ranking score (NDCG) is computed on 10 levels from top 10 to 100. The horizontal axis denotes the number of top retrieved images, and the vertical axis represents the score of NDCG.

we derivate three variations from our method to participate in the comparison. We use **SR** to represent the variation that only uses the sparse reconstruction. **SR+GR** denotes that we add the graph regularization into the objective function. **SR+RR** denotes that we add the ranking regularization into the objective function and our whole framework is represented by **SP+RR+GR**.

The comparison results are shown in Figure 4. We can observe that our method has achieved a significant improvement compared with the baselines [23,14,17] on all five types of queries. Four possible reasons may explain this situation: 1) The clean background of each image has little negative effect on training the attribute classifiers thus leads to the relative better intermediate representations. 2) The query expansion can generate the discriminative representation for the query. 3) The unified objective function can not only preserve the global ranking consistency by the ranking term but also hold the local similarity among images by the graph regularization (some examples corresponding to this point are shown in Fig. 6(a)). And 4) Few interruptions, such as viewpoint changes, occlusions and scales) in this dataset, which makes our results impressive.

3.4 CUB-200-2011 Dataset

To further test the effectiveness of our model, we experiment on a more challenging dataset CUB-200-2011 [20]. It contains 312 pre-defined binary attributes to describe a bird. The attributes can be summarized as 15 part categories: { *Beak, Belly, Throat, Crown, Tail, Back, Fore-head, Nape, Eye, Wing, Breast, Head, Leg, Body, Bird size*} and on average each part corresponds to two attributes on color or pattern. Instead of

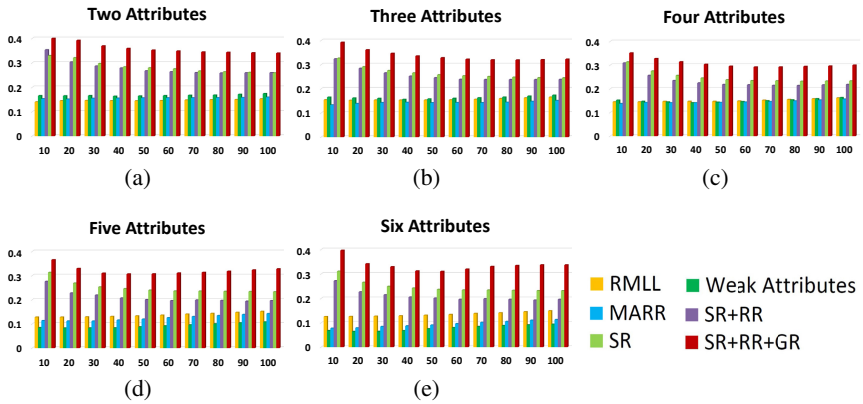
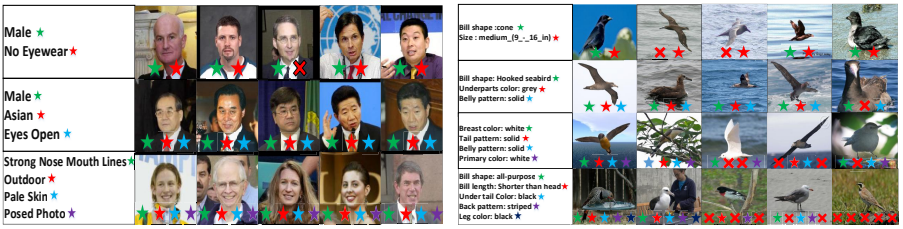


Fig. 5. The comparison results between our methods and the state-of-the-art on CUB-200-2011 dataset. From (a)-(e) are the results of five types of queries. Ranking score (NDCG) is computed on 10 levels from top 10 to 100. The horizontal axis denotes the number of top retrieved images, and the vertical axis represents the score of NDCG.

generating the queries from the binary attribute description, we develop the queries from the 15 category descriptions. We constrain that the query should contain different kinds of part categories and should include at least 30 related images. The number of queries on this dataset is 400. The parameters of this dataset is fixed as $\lambda = 10$ in the query expansion process, and $\alpha = 10, \beta = 100.5, \gamma = 0.5$ in the training.

Since there are not existing pre-trained attribute detectors, we employ a multi-label method [24] to train an multi-label attribute detector. Further, we use the image level description to describe each image: firstly we use the low level features which are provided by [5]. Three types of features including color(8), contour(128), and shape(54) are used to develop a descriptor with 216 dimensions. The comparison results of our method with the existing works are shown in Fig. 5. We observe that our proposed method achieves a significant better performance on this dataset than the other alternatives. This results further validate the robustness of our framework. While we also find that the average performance is degraded with the number of attributes growing. The main reasons are that the attribute detector has a lower recognition accuracy rate. When there are more attributes in the query, the errors accumulate. In addition, the decreasing number of positive images influences the performance of ranking SVM. Last but not least, the diverse bird appearances increase the difficulty of generating a robust classifier, even they are in the same class, which would reduce the positive effect of graph regularization. Some experiment results are shown in Fig. 6(b). We also show the qualitative retrieval example comparing with related methods as shown in Fig. 7.



(a) (b)

Fig. 6. (a) Top-5 retrieval results of our proposed approach based on different kinds of queries on LFW dataset. (b) Top-5 retrieval results of our proposed approach based on different kinds of queries of Bird dataset. The color of stars indicates the distinct attributes in the query and solid star represents the presence of corresponding attributes while the red cross states the image missing the corresponding query attribute.

Table 1. Image retrieval accuracy on shoes dataset. Comparing with [25], our proposed image retrieval framework significantly improves the accuracy.

Methods	10	30	50	70	90	Avg.
[25]	0.5811	0.4618	0.4076	0.3710	0.3417	0.4326
Our Method	0.6881	0.6033	0.5826	0.5718	0.5604	**0.6012**

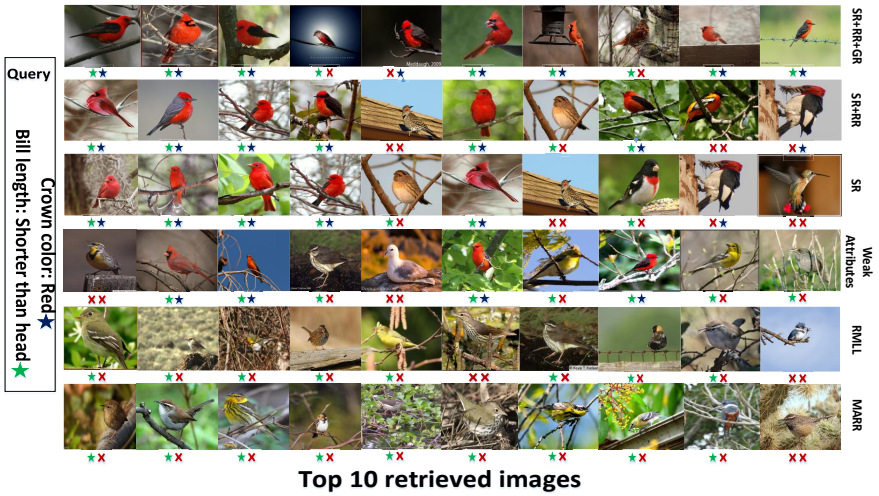


Fig. 7. Example of the retrieval results using the query "Bill length: Shorter than head" and "Crown color: Red" in the bird dataset. Top three rows show the results of our proposed methods, and the remaining rows display the results of the related work. The red cross states the image missing the corresponding query attribute. Better view in color.

3.5 Shoes Dataset

In this section, we show that our framework can be adopted to the situation that the query is an image. Different from the semantic query, we employ low level features (GIST and Color) to represent the query and corpus images. Then we reconstruct the query using the same framework as we have introduced. In total, we randomly select 100 images from the training set. We compare our ranking results with [25] whose aim was to retrieval similar images instead of ranking them. As [25] harnesses the group sparsity to retrieval images, so we group the proposed low level features into 10 parts (9 groups for GIST and 1 for Color). Mean Average Precision (**MAP**) is employed to measure the performance of different methods. The results are shown in Table 1, as can be seen, our method obtains 60.12% on average, compared with 43.26% of [25], thanks to that our method combines the ranking error into the objective function to gain a more better reconstruction results, and the graph regularization is capable to preserve the local similarity among images which helps to produce more robust results. Furthermore, comparing with [25] our proposed method add the graph and ranking regularization into the objective function which would decrease the distance between the query image and the similar reference images (Graph regularization) and increase the distance between the query image and the dissimilar reference images (Ranking regularizations).

4 Conclusion and Future Work

In this paper, we have proposed a framework for solving the multi-attribute query based image retrieval and ranking problem by minimizing both the reconstruction and the ranking errors. The proposed algorithm takes advantage of the structural sparsity of queries. To enhance the discriminative power of the query representation, the query expansion has been introduced into the framework. Compared with the state of the art image retrieval techniques with semantic queries, our proposed method has shown advanced performance. In addition, we have also applied our approach on the image retrieval with image query, our algorithm has achieved better results over the others. However, the low detection accuracy of attribute would directly affect the performance of our proposed method, how to improve the attribute prediction performance is left as our future work to further improve the performance of image retrieval and ranking.

Acknowledgments. This work was supported by National Natural Science Foundation of China (*No*.61332012), National Basic Research Program of China (2013*CB*329305), National High-tech R&D Program of China (2014*BAK*11*B*03), and 100 Talents Programme of The Chinese Academy of Sciences. X. Guo was supported by Excellent Young Talent of the Institute of Information Engineering, Chinese Academy of Sciences.

References

1. Berg, T.L., Berg, A.C., Shih, J.: Automatic attribute discovery and characterization from noisy web data. In: Daniilidis, K., Maragos, P., Paragios, N. (eds.) ECCV 2010, Part I. LNCS, vol. 6311, pp. 663–676. Springer, Heidelberg (2010)

2. Branson, S., Wah, C., Schroff, F., Babenko, B., Welinder, P., Perona, P., Belongie, S.: Visual recognition with humans in the loop. In: Daniilidis, K., Maragos, P., Paragios, N. (eds.) ECCV 2010, Part IV. LNCS, vol. 6314, pp. 438–451. Springer, Heidelberg (2010)
3. Chapelle, O., Le, Q., Smola, A.: Large margin optimization of ranking measures. In: NIPS Workshop on Learning to Rank (2007)
4. Chen, M., Zheng, A., Weinberger, K.: Fast image tagging. In: ICML (2013)
5. Duan, K., Parikh, D., Crandall, D., Grauman, K.: Discovering localized attributes for fine-grained recognition. In: CVPR (2012)
6. Jiang, Z., Lin, Z., Davis, L.: Label consistent k-svd: Learning a discriminative dictionary for recognition. TPAMI 35(11), 2651–2664 (2013)
7. Joachims, T.: Optimizing search engines using clickthrough data. In: KDD (2002)
8. Kovashka, A., Parikh, D., Grauman, K.: Whittlesearch: Image search with relative attribute feedback. In: CVPR (2012)
9. Kumar, N., Belhumeur, P., Nayar, S.: FaceTracer: A search engine for large collections of images with faces. In: Forsyth, D., Torr, P., Zisserman, A. (eds.) ECCV 2008, Part IV. LNCS, vol. 5305, pp. 340–353. Springer, Heidelberg (2008)
10. Kumar, N., Berg, A.C., Belhumeur, P.N., Nayar, S.K.: Attribute and simile classifiers for face verification. In: ICCV (2009)
11. Lin, Z., Ding, G., Hu, M., Wang, J., Ye, X.: Image tag completion via image-specific and tag-specific linear sparse reconstructions. In: CVPR (2013)
12. Liu, J., Ji, S., Ye, J.: Multi-task feature learning via efficient l2,1-norm minimization. In: UAI (2009)
13. Liu, S., Song, Z., Liu, G., Xu, C., Lu, H., Yan, S.: Street-to-shop: Cross-scenario clothing retrieval via parts alignment and auxiliary set. In: CVPR (2012)
14. Petterson, J., Caetano, T.: Reverse multi-label learning. In: NIPS (2010)
15. Rastegari, M., Diba, A., Parikh, D.: Multi-attribute queries: To merge or not to merge? In: CVPR (2013)
16. Scheirer, W., Kumar, N., Belhumeur, P.N., Boult, T.E.: Multi-attribute spaces: Calibration for attribute fusion and similarity search. In: CVPR (2012)
17. Siddiquie, B., Feris, R.S., Davis, L.S.: Image ranking and retrieval based on multi-attribute queries. In: CVPR (2011)
18. Tsochantaridis, I., Joachims, T., Hofmann, T., Altun, Y.: Large margin methods for structured and interdependent output variables. JMLR 6, 1453–1484 (2005)
19. Wang, C., Yan, S., Zhang, L., Zhang, H.J.: Multi-label sparse coding for automatic image annotation. In: CVPR (2009)
20. Welinder, P., Branson, S., Mita, T., Wah, C., Schroff, F., Belongie, S., Perona, P.: The caltech-ucsd birds-200-2011 dataset. California Institute of Technology, CNS-TR-2011-001 (2007)
21. Wright, J., Yang, A.Y., Ganesh, A., Sastry, S.S., Ma, Y.: Robust face recognition via sparse representation. TPAMI 31(2), 210–227 (2009)
22. Wu, L., Jin, R., Jain, A.K.: Tag completion for image retrieval. TPAMI 35(3), 716–727 (2013)
23. Yu, F., Ji, R., Tsai, M.H., Ye, G., Chang, S.F.: Weak attributes for large-scale image retrieval. In: CVPR (2012)
24. Zhang, M., Zhou, Z.: Ml-knn: A lazy learning approach to multi-label learning. PR 40(7), 2038–2048 (2007)
25. Zhang, S., Huang, J., Li, H., Metaxas, D.N.: Automatic image annotation and retrieval using group sparsity. TSMC, Part B, 838–849 (2012)
26. Zheng, M., Bu, J., Chen, C., Wang, C., Zhang, L., Qiu, G., Cai, D.: Graph regularized sparse coding for image representation. TIP 20(5), 1327–1336 (2011)