# Negative-Voting and Class Ranking Based on Local Discriminant Embedding for Image Retrieval

Mei-Huei Lin, Chen-Kuo Chiang and Shang-Hong Lai
Department of Computer Science, National Tsing Hua University, Hsinchu, Taiwan, R.O.C.
E-mail: h85164@hotmail.com, ckchiang@cs.nthu.edu.tw, lai@cs.nthu.edu.tw

***Abstract*—In this paper, we propose a novel image retrieval system by using negative-voting and class ranking schemes to find similar images for a query image. In our approach, the image features are projected onto a new feature space that maximizes the precision of image retrieval. The system involves learning a projection matrix for local discriminant embedding, generating class ordering distribution from a negative-voting scheme, and providing image ranking based on class ranking comparison. The evaluation of mean average precision (mAP) on the Holidays dataset shows that the proposed system outperforms the existing retrieval systems. Our methodology significantly improves the image retrieval accuracy by combining the idea of negative-voting and class ranking under the local discriminant embedding framework.**

## I. Introduction

In recent years, the amount of digital information has increased tremendously because of the surge of internet. An important type of information is multimedia, such as image and video. Multimedia retrieval plays an important role in efficient search of multimedia from a huge database. A number of state-of-the-art image and video retrieval approaches adopted the standard bag-of-words model initially introduced by Sivic et al. [1]. Inspired by Google's document retrieval technique, they developed the Bag-of-visual-words method to deal with the image/video retrieval problem. The objective of the bag-of-visual-words approach is to quantize the feature point descriptors into clusters, and represent images with histograms of quantized features.

The Bag-of-visual-words model works generally well for image retrieval. However, feature quantization degenerated the discriminative power of features. The intrinsic similarity and dissimilarity between the features are abandoned because only the histograms of the quantized features for images are used. In order to maintain the relationship between data points, this paper utilizes the local discriminant embedding [2] method. The advantage of local discriminant embedding is to map data points from the original feature space into a lower-dimensional subspace while keeping similar data points closer and pushing dissimilar data points away. Local discriminant embedding is usually employed for classification problems. So, it requires additional steps to modify the classification techniques for the retrieval problem. In this paper, a negative-voting retrieval system is proposed based on the local discriminant embedding technique.

There are many existing classification techniques. Principle component analysis (PCA), or equivalently the Karhunen-Loeve transform, is widely used for linear dimensionality reduction. Linear discriminant analysis (LDA) considers the geodesic distances between data points, and is shown to be a useful tool for feature extraction and classification. The LDA framework seeks to dissociate each class from one another, and not only retains the similarity, but also takes the dissimilarity into consideration. While maintaining the original neighboring relations for data points of the same class is important, it is also crucial to differentiate and separate data points of different classes after the embedding.

Image retrieval regards every image as an independent individual. Exploiting the class information is helpful to the retrieval problem. In our approach, each feature finds one most likely category by local discriminant embedding technique. For all features of an image, the statistical analysis is applied to obtain the category distribution of features. Assume similar images have similar feature representation. By comparing the feature distribution, the rank of similar images can be determined for the image database. To make the results more discriminative, a negative voting scheme is proposed to reduce the influence of outliers. It penalizes most dissimilar category information in statistical analysis. After the negative voting, images are associated with a distribution from the most likely class to the most unlikely class. Since similar images have similar class distributions, the ranking list can be obtained by comparing the class distribution.

The Holidays dataset [3] is utilized to evaluate the proposed image retrieval system with the mean average precision (mAP). The results show that our approach can improve the precision of the image retrieval significantly. The main contributions of this paper are two-fold:

1) Incorporate class information based on the local discriminant embedding technique along with dissimilar information into the image retrieval system.
2) A negative voting scheme is proposed to reduce the influence of outliers and yields better ranking results.

The rest of the paper is organized as follows. We review the related works in the Section II. Then, we describe the local discriminant embedding technique in Section III. The negative voting and the generation of ranking list based on class distribution is presented in Section IV and Section V, respectively. Next, we describe our system framework for image retrieval in Section VI. Experimental comparison of our method to other state-of-the-art systems is given in Section

VII. Finally, Section VIII concludes this paper.

## II. Related Work

Due to the rapidly growing number of online images, modern image retrieval systems need to handle these resources effectively and efficiently. One way is to recognize the class label of the query image prior to postprocessing or similarity matching steps. This enables the content-based search in a reduced image collection and find images with similar class labels. In [4], the image collection was first categorized by using multi-class Support Vector Machine (SVM). Then, online category prediction of the query image was applied to eliminate images from irrelevant classes. Last, category-specific feature weights are exploited in a linearly combined similarity matching function. In [5], the classification is applied in the pixel level. By classifying image pixels to either border or interior categories, a logarithmic distance can be applied easily for comparing histogram-based features. To deal with large numbers of high-dimensional features, dimension reduction techniques are often applied. Reduction can be applied to the image as a whole [6], [7] or to feature vectors [8]. Yang [9] showed an image classification scheme that relies on the distance-preserving method that projects the data down to a lower-dimensional space by using the distances between a point and some of its near neighbors. It can successfully project data even when the data points are not evenly distributed.

Multidimensional projections are commonly applied to multidimensional data by projecting data defined in an $p$-dimensional space into a $q-$dimensional space where $q \ll p$. The data relationship in the projected space should be preserved as that defined in the original space. Two major groups are defined according to the functions employed: linear projection function and non-linear projection function. Linear projection defines data by a new orthogonal basis of lower dimension. The widely known method is Principal Component Analysis (PCA) [10]. Nonlinear techniques attempt to minimize a function of the information loss after projection, such as Curvilinear Component Analysis (CCA) [11] and ISOMAP [7].

## III. Local Discriminant Embedding

Since exploiting the class relation between a data point and its neighbors is not easy, the local discriminant embedding [2] method achieves excellent performance for the classification problem. To preserve the relation for features of the same class is a major goal in such subspace methods. It is also important to push neighboring data points from different classes away after the projection step to enhance the discriminant ability. Local discriminant embedding incorporates the class information into the construction of projection matrices. It distinguishes the class of a query image in a low-dimensional Euclidean space by maximizing the margin between classes. The key ideas of local discriminant embedding are listed as follows:

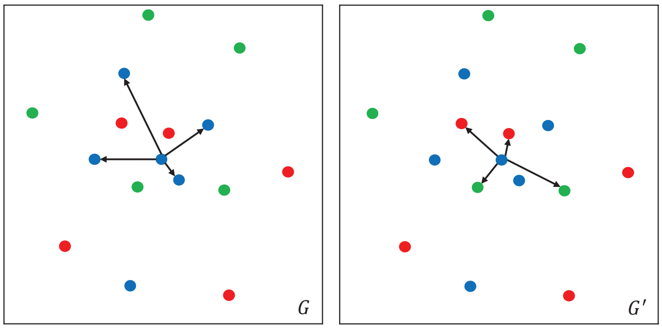


Fig. 1. An example of constructing neighborhood graphs. The same color means the same class label. In Graph $G$, four nearest neighbors from the same class with blue color are connected whereas four nearest neighbors from different classes are connected in Graph $G'$.

1) Locally evaluate the similarity and dissimilarity in neighborhood area.
2) The embedding is in the form of linear projection, which is obtained by finding the generalized eigenvectors.
3) Local discriminant embedding is formulated as solving an optimization problem.

Local discriminant embedding can be broken down into three steps. Suppose we have $m$ data points in $n$ dimensional space $\{x_i \mid x_i \in \mathbf{R}^n\}_{i=1}^m$ and the corresponding class label $\{y_i \mid y_i \in \{1, 2, ..., P\}\}_{i=1}^m$ in $P$ classes. The data matrix can be rewritten as $\mathbf{X} = [x_1 x_2 ... x_m] \in \mathbf{R^{n \times m}}$. Next, we introduce the method step by step.

### A. Construct neighborhood graphs

Denote $G$ and $G^{'}$ to be two graphs describing the local neighborhood relation for a feature point. $G$ is a graph that represents the similarity of neighboring data points from the same class. $G^{'}$ is a graph of the neighboring data points from different classes. To build $G$, consider the pair of features $x_i$ and $x_j$ from the same class, namely, $x_i$ and $x_j$ which have the same class label $y_i = y_j$. If $x_j$ is one of $x_i^{'}sk$-nearest neighbor, the edge between $x_i$ and $x_j$ is constructed in $G$. In the construction of $G^{'}$, we consider the pair of $x_i$ and $x_j$ with different class labels. In other words, $y_i \neq y_j$. Connect $x_i$ and $x_j$ if $x_j$ is one of $x_i^{'}s$ $k$-nearest neighbors from different classes.

### B. Choose the weights

When constructing the graph $G$, the affinity matrix $W$ can be built at the same time. Each element $w_{ij}$ of matrix $W$ refers to the weight of the edge between feature points $x_i$ and $x_j$. If $x_i$ and $x_j$ are connected to each other, the element $w_{ij}$ will be assigned to one, i.e. $w_{ij}$ = 1, and $w_{ij}$ = 0, otherwise. We also compute the weight matrix $W^{'}$ of $G^{'}$ in the same way. It is obviously that $W$ and $W^{'}$ are $m \times m$, sparse and symmetric matrices. Figure 1 depicts an example of constructing neighboring relation for $G$ and $G^{'}$.

### C. Compute the projection matrix

Unlike PCA and LDA, local discriminant embedding explores the local relations between neighboring feature points. One of the most important goal of this method is keeping neighboring points close if they have same class label, whereas pushing away points from different classes. Given the data point $x$, the projected data $z$ can be obtained by the linear project matrix $V$, $\mathbf{z} = V^T\mathbf{x}$, where $V$ is $n \times l$ matrix with $l \ll n$. Through the projection, the relations of neighboring points can be well preserved via the affinity matrix $W$ and $W^{'}$ . It can be formulated as a constrained optimization problem as follows:

$$\begin{aligned} Maximize\ J(V) &= \sum_{i,j} \parallel V^T x_i - V^T x_j \parallel^2 w_{ij}^{'} \\ subject\ to\ &\sum_{i,j} \parallel V^T x_i - V^T x_j \parallel^2 w_{ij} = 1. \end{aligned} \tag{1}$$

The above formulation maximizes the distance of data points from different classes after projection and keeps the data distance within the same class. The formulation which exploits the class and neighbors information can be rewritten as:

$$\begin{aligned} J(V) &= \sum_{i,j} \parallel V^T x_i - V^T x_j \parallel^2 w_{ij}^{'} \\ &= tr\{V^T \sum_{i,j}((x_i - x_j) w_{ij}^{'} (x_i - x_j)^T) V\} \\ &= 2tr\{V^T X (D^{'} - W^{'}) X^T V)\}, \end{aligned} \tag{2}$$

where $X$ is the data matrix and $D^{'}$ is a diagonal matrix with $d_{ii}^{'} = \sum_j w_{ij}^{'}$. Eq.1 can be rewritten as:

$$\begin{aligned} Maximize\ J(V) &= 2tr\{V^T X (D^{'} - W^{'}) X^T V\} \\ subject\ to\ &2tr\{V^T X (D - W) X^T V)\} = 1. \end{aligned} \tag{3}$$

The solution of the above the optimization problem is to solve the generalized eigenvector for the following equation [2]:

$$X(D^{'} - W^{'}) X^T v = \lambda X (D - W) X^T v. \tag{4}$$

The generalized eigenvectors $\mathbf{v_1}, \mathbf{v_2}, \cdots, \mathbf{v}_l$ correspond to the $l$ largest eigenvalues, and $V = [\mathbf{v_1}\ \mathbf{v_2}\ \ldots\ \mathbf{v}_l]$. After deriving the projection matrix $V$, the classification becomes straightforward. For any test feature point $\bar{x} \in \mathbf{R^n}$, it can be projected onto a new space. The class label $y_i$ can be predicted by the label of data point $z$ which minimizes $\parallel z_i - \bar{z} \parallel$.

## IV. Negative Voting

Local discriminant embedding finds the projection matrix $V$. Feature point can be projected onto the new space by $z_i = V^T x_i$. In our approach, all training data points are projected. Then, the class mean is calculated. Given a feature point of a query image, the class label is decided by the minimal distance to the class mean. Ideally, feature points of images belong to the same class stay closer. Unfortunately, there are outliers, such as meaningless or irrelevant feature points. To reduce the influence of outliers, a negative voting method is exploited to estimate robust class labels. The idea is to vote for the most likely class, and vote with negative scores for unlikely classes as well.

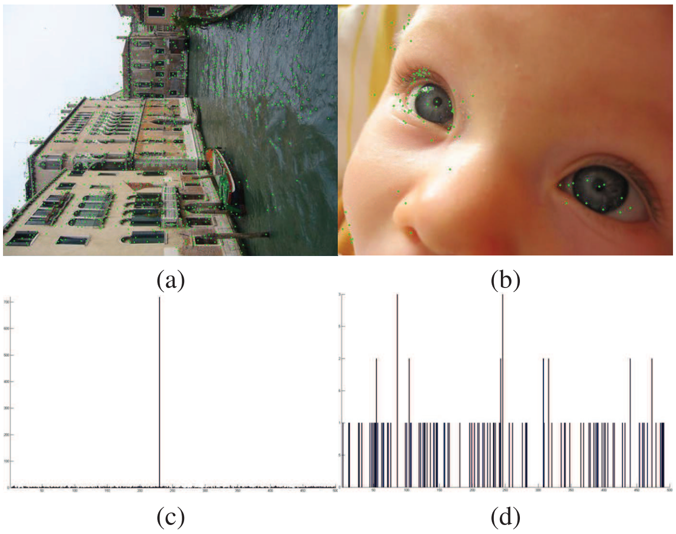


Fig. 2. (a) and (b) show the positions of features in No.229 and No.381 query images, respectively. (c) and (d) are the voting results corresponding to No.229 and No.381 query images. The voting outcome of No.381 query indicates the local discriminant embedding fails. Our proposed method utilizes negative voting to improve the robustness of vote distribution, which is used for image retrieval.

Voting negative classes penalizes the far away classes, providing more accurate information of class voting. Figure 2 shows the numbers of votes for the classes. An ideal case is given by the voting number generated by query image 229. Only one peak exists in the voting results of Fig.2 (c). The decision for the class can be made according to the highest votes easily. The voting result of query image 381 in figure 2 (d) shows there are two peaks for class decision. In the case of multiple peaks, the class decision becomes difficult. By voting for the near class and far away classes, it is helpful to break a tie and reduce the impact of outliers. In the implementation, we can simply subtract the number of positive votes by the negative votes. A weighting factor can also be applied to the negative votes before the subtraction.

After obtaining the number of votes for each class, intuitively we can retrieve similar images by comparing two histograms of vote numbers. Similar images should have similar distributions for the class voting. Considering different number of feature points for each image and the various number of outliers, the voting results are not used directly for image retrieval in this paper. We utilize the order of classes from the highest votes to the most unlikely class as a new feature to represent the image. Specifically, the voting results are first transformed to class order based on the number of votes. Class order is descending with the most likely class in the front. We will introduce the details of class distribution ranking in the next section.

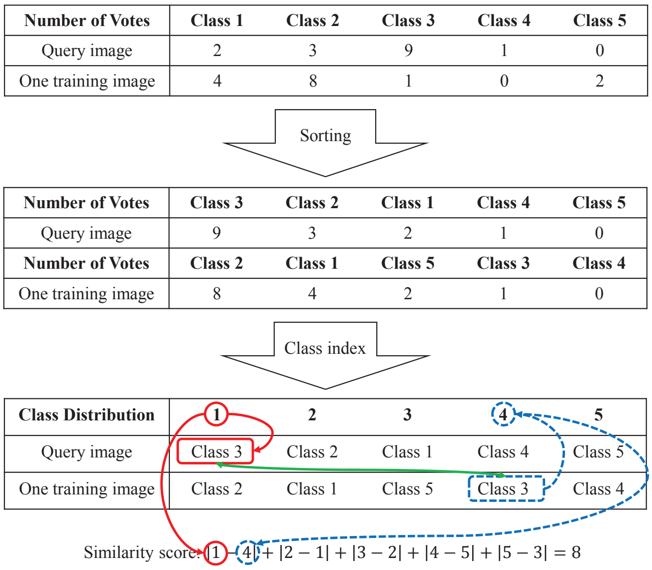


Fig. 3. An example to explain our ranking method. First, the numbers of votes are sorted to obtain the class order distribution. In step two, from the class distributions of training images, we find the class with the most likely class distribution with that of the query image. In this case, the query image seems most probably belong to class 3, which is the first place in the query class distribution. Class 3 is the fourth place in the class order of training data. The colors of lines indicate the sequence of this step, finding order in query is red, looking for the corresponding class is green, and seeking the index of this class in training image is blue. Since the similarity score between query and training image defined as the difference of two indices, we repeat the above step to accumulate the differences of each class index pair.

## V. Class Distribution Ranking

Our proposed method utilizes class order distribution as a new global feature. Class distribution shows the possibility of which class the image belongs to. If two images share the same content, we assume the class order of these images are similar. Based on the assumption, a feature based on class order is proposed for image retrieval.

Figure 3 illustrates an example to calculate the similarity scores of two images. Through the local discriminant embedding projection, each feature finds its closest class. Assume there are five classes. The top table shows the number of votes for five classes of one query image and one image from the database. The first step is to sort the voting results and obtain the class order distributions. Then, we compare the indices of the same class in two distributions. In this example, the most possible class for the query image is class 3 and class 3 in the class distribution of the training image is located at the fourth place. By calculate the distance between the class indices $\mid 1-4 \mid$, the similarity can be calculated by summing the differences over the total class numbers:

$$\sum_{c=1}^{\sharp class} g(I_c^q, I_c^t) \tag{5}$$

where function $g$ represents the comparing method, and $I$ is the index in class distribution (the superscript $q$ means in query class distribution, and $t$ means in training class distribution). To compute the similarity between images, we can simply accumulate the distance of indices as follows:

$$g_1(I_c^q, I_c^t) = \mid I_c^q - I_c^t \mid . \tag{6}$$

The above equation considers only the absolute value of indices distance to evaluate the similarity scores. Considering the class indices closer to the beginning of the distribution list are more important than those in the end of the list, the weighting factor can be given as follows:

$$g_2(I_c^q, I_c^t) = \mid I_c^q - I_c^t \mid \times e^{-I_c^q}. \tag{7}$$

Note that if the weight is set very high, it leads to the situation that the results may be dominated by a few classes. In Section VII, several experiments are provided by using different ranking methods and show the corresponding results.

## VI. System Framework of Image Retrieval

Most image retrieval systems retrieve images based on the similarity. Our method not only exploits the similarity but also penalizes the dissimilarity. In this section, we will introduce the components and the details of our image retrieval system.

Our system consists of two processes: training and query process. Figure 4 depicts the overall system flow. The solid blue line represents the training procedure and the red dotted line indicates the query process. We exploit the class labels of training images. If the labels are not provided, a pre-classification step is required. Local feature, such as SIFT [12], is a popular choice since global features usually cannot support object-level matching. On the other hand, utilizing local features to represent images is also a reasonable choice for the voting scheme in our approach. We choose SIFT as the local feature because of its characteristics of invariance to translation, rotation and scaling. As the image database may have hundreds of thousands SIFT features, the second step is to separate them into different clusters. K-means clustering [13] is used in training process to find the cluster centers. Then, each feature is associated with a nearest cluster center.

After clustering, the next step is to perform LDE. Note that, the dimension of the affinity matrix in LDE is the total number of feature points $\times$ total number of feature points. Generally, it is not possible to calculate the matrix for the whole dataset. Therefore, LDE is applied within each cluster. Then, the class mean are saved for each class in each cluster. By using LDE, every training data point can be projected onto a new feature space and find the nearest class by comparing to the class mean. Feature points vote for the nearest class and the further classes via negative voting. The distribution of class order is used to construct a new feature to represent the image. In the last step of training, we represent all the images in the database with the associated class distributions, which are compared with the class distribution of the query image in the testing phase.

We summarize the steps for image retrieval for a query image as follows:

1) Extract SIFT features of the query image.

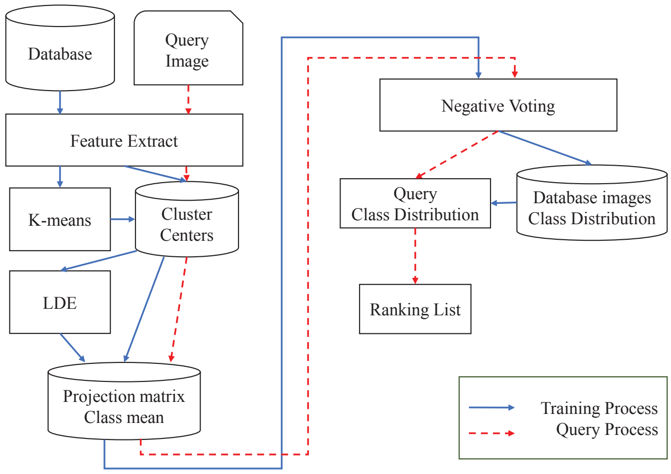


Fig. 4. Architecture of the proposed image retrieval system approach based on local discriminant embedding and negative voting. Our system consists of the training flow and query flow, which are denoted by blue solid arrow and rad dotted arrow, respectively.

2) For each feature points, find its cluster by comparing to the K-means centroids.
3) Project the data point into subspace using the cluster-specific projection matrix.
4) Vote for the nearest class and the farthest classes.
5) Obtain the class distribution and calculate the similarity scores to obtain the ranking list.

## VII. Experimental Results

We evaluate our method using the Holidays database [3]. Sample images from this dataset are shown in Figure 5. Although Holidays dataset is not a large-scale dataset, it contains a lot of variations in images, such as lighting, viewpoint changing and rotations, as depicted in Figure 6. Therefore, it is suitable for evaluation of image retrieval systems. In the image retrieval problem, there are many evaluation criteria, such as precision, recall, and ROC curve. We utilizes the mean average precision (mAP) as the evaluation criterion. The range of mAP is between 0 and 1. Higher values mean better retrieval performance.

SIFT [12] is used as the feature descriptor in our experiments. The feature dimension is 128. We separated feature points into 975 clusters by $K - means$. Local discriminant embedding keeps the relation of neighborhood and reduces the dimension to $l$. We set $l$ = *60* in our implementation. Holidays dataset contains 500 image groups. Each image group represents a distinct scene or object. The first image of each group is the query image and the correct retrieval results are the other images of the group. The above parameter settings are used for the experiments of negative voting and class distribution ranking in the proposed image retrieval system.

### A. Performance of Negative Voting

By LDE, every feature finds the nearest class and votes to the class. The results of voting are likely to have no peak

Fig. 5. Sample images from the Holidays database.

(a) (b) (c)

Fig. 6. Holidays database contains rich variability, such as lighting (a), viewpoint change (b), image rotation (c), out of focus and image size alteration.

or multiple peaks. Sometimes, it generates a wrong peak. To resolve this problem, a negative voting scheme is introduced. In this experiment, we evaluate the system by evaluating the performance of image retrieval with and without using the negative voting scheme.

Table 1 shows the retrieve results in mAP and Figure 6 is the top three images from the ranking list for querying image No. 376 by using different negative voting schemes. The query image No. 376 has three ground truth images in the dataset. Note that, the quality of query image No. 376 is poor which is blurred. In the experiment without using the negative voting, the mAP is only 18.79%. Only one of its ground truth images is retrieved in the top three of ranking list.

The second set retrieves images using negative voting which

TABLE I
The performances of image retrieval with and without using the negative voting method by using the mAP criterion.

| method | mAP(%) |
|---|---|
| Without negative voting | 18.78 |
| Negative voting for one farthest class | 27.74 |
| Negative voting for bottom half of far classes | 89.28 |

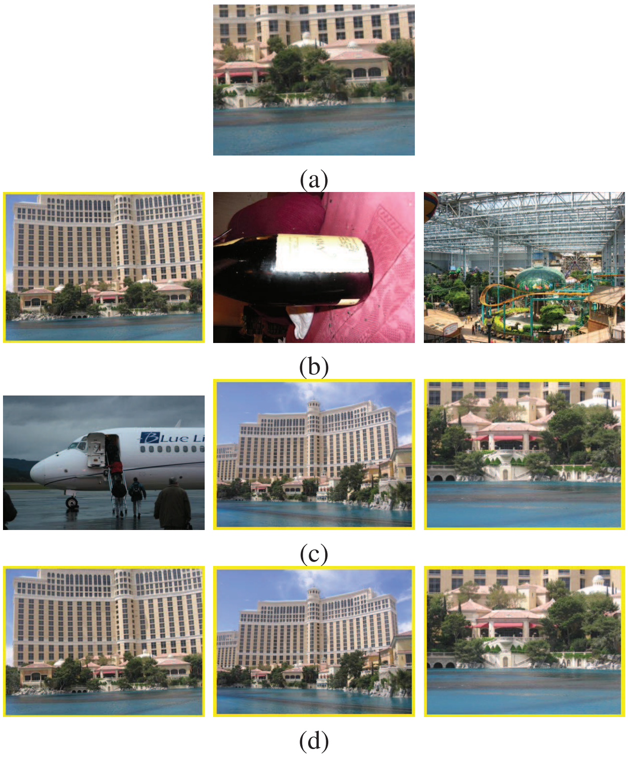


Fig. 7. Retrieval results of negative voting experiment. (a) No. 376 query image. (b) the retrieval results of first three images without negative voting, (c) Retrieval results using negative voting by voting only one farthest class, and (d) negative voting the bottom half of far-away classes. Images, which have yellow frame, have the correct class label as query image No.376.

votes one farthest class. The mAP is improved to 27.74% and there are two ground truth images in the top-three ranking list. The last group uses the negative voting for half of the farthest classes. The performance is significantly enhanced by using this method, and the mAP reaches up to 89.28%. From this experiment, our approach achieves significant improvement by combining the negative voting technique into the image retrieval system.

## B. Performance of Class Distribution Ranking

Although we utilize voting distribution to represent an image, it also needs a good way to process the similarity score. In this experiment, we compare the class ranking results by using function $g$ along with decreasing weighting values for further ranks. Table 2 demonstrates the results by using different methods, and Figure 7 depicts the corresponding results of the ranking list. We use image No.376 as the query image. The images with yellow frame are the ground truth images. Three methods are used to calculate the similarity score. First, we sum the differences of class ranks for all classes, as eq. (6). Second, we associate higher weights to the classes located in the beginning of the list, as shown in eq. (7). Last, we calculate the similarity score by the following function:

$$g_3(I_c^q, I_c^t) = (\mid I_c^q - I_c^t \mid \times e^{-I_c^q})^2. \tag{8}$$

TABLE II

THE PERFORMANCES OF THE CLASS DISTRIBUTION RANKING APPROACH BY USING THE MAP EVALUATION CRITERION.

| method | mAP(%) |
|---|---|
| $\mid I_c^q - I_c^t \mid$ | 38.95 |
| $\mid I_c^q - I_c^t \mid \times e^{-I_c^q}$ | 89.28 |
| $(\mid I_c^q - I_c^t \mid \times e^{-I_c^q})^2$ | 89.17 |

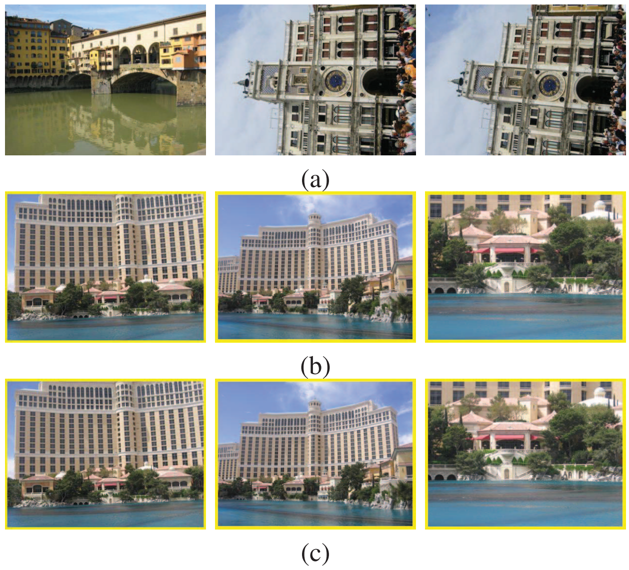


Fig. 8. Results of different ranking methods. (a) results using equal-weight class ranking method, (b) results using the weighted class ranking method, and (c) the results using weighted square ranking method.

From Table 2, we can find the results of weighting methods outperform the method with equal weighting. The third method does not improve the second method. In Figure 8, the top three list of retrieval results are incorrect images retrieved by using equal weighting. By using an appropriate weighting function, method 2 and method 3 correctly retrieve all the ground truth images.

## C. Evaluating Retrieval System

To evaluate the performance of our image retrieval system, we compared to four methods. First, the hamming embedding method [3] is used as baseline method. It utilizes Bag-of-feature model and refines the matched visual words based on hamming embedding and weak geometric consistency criterion. As a result, the method improves the accuracy of object retrieval based on Bag-of-feature model greatly. Second, the method [14] utilizes query-adaptive criterion to weight the descriptor matches, and refines the similarity measure between descriptors based on reciprocal nearest neighbor (RNN). The methods of the other two papers include the spatial information to improve the performances of vocabulary tree [15] and bag-of-words [16]. Table 3 presents the results of mAP values by using the above methods. Our method outperforms the baseline method and provides higher mAP than the other state-of-the-art methods.

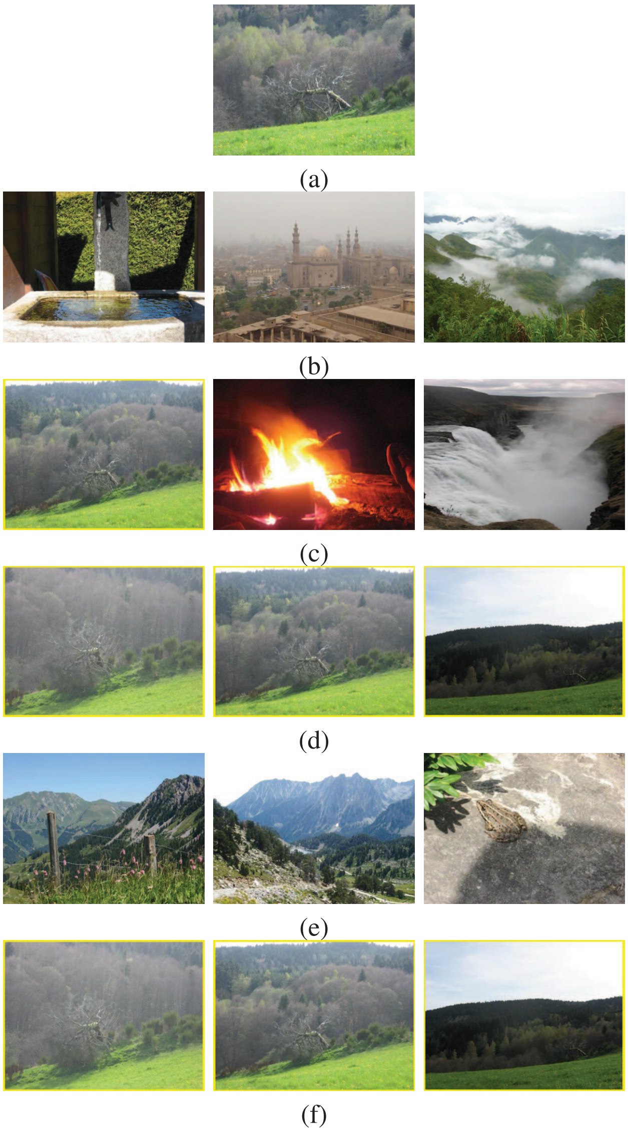


Fig. 9. Sample results of our retrieval system using query image No.486. (a) the query image. (b), (c), (d) correspond to the results without negative voting, with negative voting for one most dissimilar class, and with negative voting for half of dissimilar classes, respectively. (d), (e), (f) correspond to the results using three different class ranking functions, $g_2$, $g_1$, and $g_3$, respectively.

TABLE III
THE PERFORMANCES OF STATE-OF-THE-ART IMAGE RETRIEVAL METHODS.

| method | mAP(%) |
|---|---|
| HM & WGC [3] | 75.07 |
| EDD [14] | 86.80 |
| CW for VT [15] | 78.00 |
| SCSM & R-NN [16] | 76.20 |
| proposed method | **89.28** |

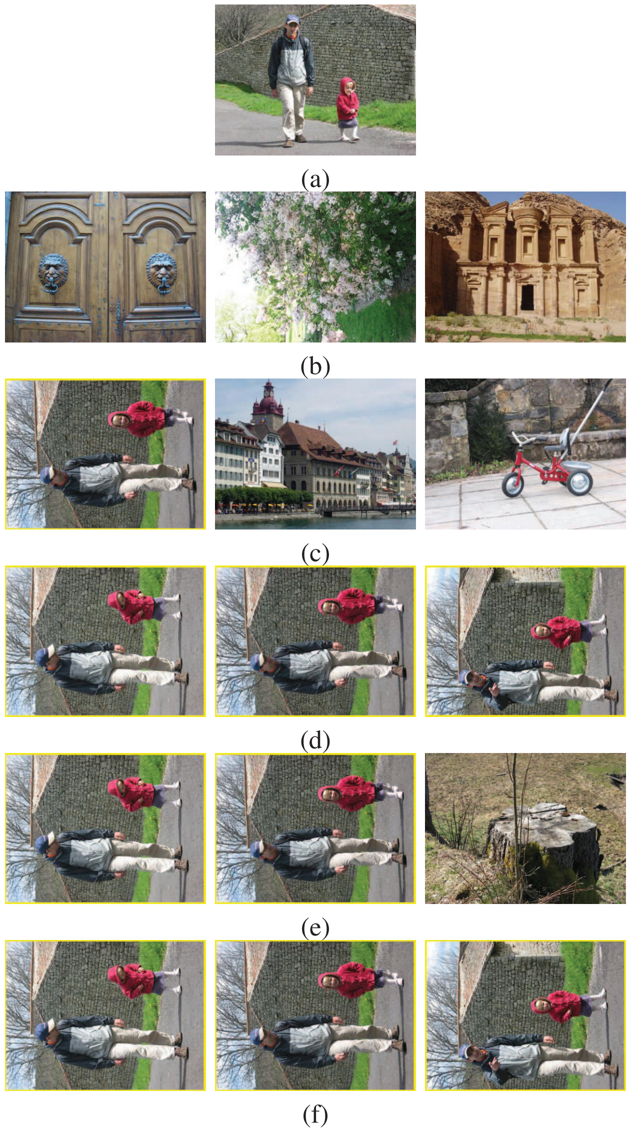


Fig. 10. Sample results of our retrieval system using query image No.460. (a) the query image. (b), (c), (d) correspond to the results without negative voting, with negative voting for one most dissimilar class, and with negative voting for half of dissimilar classes, respectively. (d), (e), (f) correspond to the results using three different class ranking functions, $g_2$, $g_1$, and $g_3$, respectively.

## VIII. Conclusion

In this paper, a novel image retrieval system is proposed based on the local discriminant embedding. By utilizing negative voting and weighted class distribution ranking algorithms, the retrieval accuracy is improved greatly based on the Bag-of-feature model. Our system considers not only similar classes, but also penalizes dissimilarity classes. From the experimental results, our system works better than the other state-of-the-art retrieval systems.

Since this paper utilizes the classification method to deal with the image retrieval problem, the system is separated into training and query stage. Currently, the training is very time-consuming. The future work is to speedup the training phase and extend this image retrieval framework to other big-data problems.

## REFERENCES


[1] J. Sivic and A. Zisserman, "Video google: A text retrieval approach to object matching in videos," in *ICCV*, 2003, pp. 1470–1477.

[2] H.-T. Chen, H.-W. Chang, and T.-L. Liu, "Local discriminant embedding and its variants," in *CVPR (2)*, 2005, pp. 846–853.

[3] H. Jegou, M. Douze, and C. Schmid, "Hamming embedding and weak geometric consistency for large scale image search," in *ECCV (1)*, 2008, pp. 304–317.

[4] M. M. Rahman, S. Antani, and G. R. Thoma, "A biomedical image retrieval framework based on classification-driven image filtering and similarity fusion," in *ISBI*, 2011, pp. 1905–1908.

[5] R. O. Stehling, M. A. Nascimento, and A. X. Falcão, "A compact and efficient image retrieval approach based on border/interior pixel classification," in *CIKM*, 2002, pp. 102–109.

[6] S. T. Roweis and L. K. Saul, "Nonlinear dimensionality reduction by locally linear embedding," *SCIENCE*, vol. 290, pp. 2323–2326, 2000.

[7] J. B. Tenenbaum, V. Silva, and J. C. Langford, "A global geometric framework for nonlinear dimensionality reduction," *Science*, vol. 290, no. 5500, pp. 2319–2323, 2000.

[8] B. Moghaddam, Q. Tian, N. Lesh, C. Shen, and T. S. Huang, "Visualization and user-modeling for browsing personal photo libraries," *International Journal of Computer Vision*, vol. 56, p. 2004, 2004.

[9] L. Yang, "Distance-preserving projection of high-dimensional data for nonlinear dimensionality reduction," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 9, pp. 1243–1246, Sep. 2004.

[10] I. T. Jolliffe, *Principal Component Analysis*. Springer-Verlag, 1986.

[11] P. Demartines and J. Herault, "Curvilinear component analysis: a self-organizing neural network for nonlinear mapping of data sets," 1996.

[12] D. G. Lowe, "Object recognition from local scale-invariant features," in *Proceedings of the International Conference on Computer Vision-Volume 2 - Volume 2*, ser. ICCV '99. Washington, DC, USA: IEEE Computer Society, 1999, pp. 1150–. [Online]. Available: http://dl.acm.org/citation.cfm?id=850924.851523

[13] J. B. MacQueen, "Some methods for classification and analysis of multivariate observations," in *Proc. of the fifth Berkeley Symposium on Mathematical Statistics and Probability*, L. M. L. Cam and J. Neyman, Eds., vol. 1. University of California Press, 1967, pp. 281–297.

[14] H. Jegou, M. Douze, and C. Schmid, "Exploiting descriptor distances for precise image search," in *INRIA research report*, 2011.

[15] X. Wang, M. Yang, T. Cour, S. Zhu, K. Yu, and T. X. Han, "Contextual weighting for vocabulary tree based image retrieval," in *ICCV*, 2011, pp. 209–216.

[16] X. Shen, Z. Lin, J. Brandt, S. Avidan, and Y. Wu, "Object retrieval and localization with spatially-constrained similarity measure and k-nn re-ranking," in *CVPR*, 2012, pp. 3013–3020.