

Multi-Manifold Ranking: Using Multiple Features for Better Image Retrieval

Yang Wang^{†‡}, Muhammad Aamir Cheema[†], Xuemin Lin[†], and Qing Zhang^{‡†}

[†]The University of New South Wales, Sydney, Australia
{wangy, macheema, lxue}@cse.unsw.edu.au

[‡]Australian E-Health Research Center
Qing.Zhang@csiro.au

Abstract. Manifold Ranking (MR) is one of the most popular graph-based ranking methods and has been widely used for information retrieval. Due to its ability to capture the geometric structure of the image set, it has been successfully used for image retrieval. The existing approaches that use manifold ranking rely only on a single image manifold. However, such methods may not fully discover the geometric structure of the image set and may lead to poor precision results. Motivated by this, we propose a novel method named **Multi-Manifold Ranking** (MMR) which embeds multiple image manifolds each constructed using a different image feature. We propose a novel cost function that is minimized to obtain the ranking scores of the images. Our proposed multi-manifold ranking has a better ability to explore the geometric structure of image set as demonstrated by our experiments. Furthermore, to improve the efficiency of MMR, a specific graph called anchor graph is incorporated into MMR. The extensive experiments on real world image databases demonstrate that MMR outperforms existing manifold ranking based methods in terms of quality and has comparable running time to the fastest MR algorithm.

Key words: Image retrieval, integrated features, manifold ranking

1 Introduction

Traditional image retrieval techniques rely on the semantic labels attached to the images such as image annotations [13] and tags [7]. However, a severe drawback of such techniques is that the manual labelling is laborious, expensive and time-consuming. Another disadvantage is that such techniques do not consider the content of the images and this may lead to poor results especially if the quality of the labelling is poor.

To address the issues mentioned above, content-based image retrieval (CBIR) [10, 5, 12] may be used which utilizes the low-level features (e.g., color, shape, texture) for image retrieval. These low-level features can be extracted automatically and remain consistent for each image in contrast to the manually attached labels. However, it is difficult to choose an ideal descriptor for the images because the low-level features may not represent the same semantic concepts. For example,

two images having similar color visualization may have totally different semantic meanings (e.g., a green apple and a tennis ball as shown in Fig. 1). This is one of the main challenges CBIR needs to address.

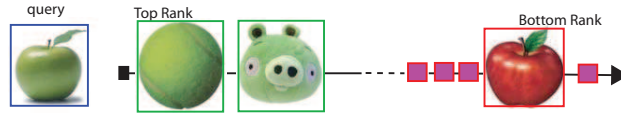


Fig. 1. If only the color feature is used, the most relevant results include a green tennis ball and a green angry bird instead of the red apple. Hence, a single feature may not provide desired results.

To address this challenge, He *et al.* [4] used manifold ranking that uses low-level features as well as the intrinsic structure of the images. The basic idea behind the manifold ranking is as follows. A weighted graph is constructed where the vertices represent the images and, for each vertex, its near by vertices are connected to it by weighted edges. The queries are assigned a positive ranking and the remaining vertices are ranked with respect to the queries. The vertices spread their ranking scores to their neighbors via the weighted graph. The spread process is repeated until convergence. This approach has been shown to yield better retrieval results because it utilizes the intrinsic structure of the image set. Xu *et al.* [18] proposed a faster manifold ranking approach that uses anchor graphs [8] to approximate the original graph and provides the results of similar quality.

The above mentioned manifold ranking techniques use a single feature. In other words, these techniques utilize the intrinsic structure of the images based only on a single feature. The ranking based on the single manifold may have low precision especially if the selected feature is not very representative. Motivated by this, in this paper, we propose a technique called multi-manifold ranking (MMR) that ranks the images by considering multiple manifolds each constructed using a different feature. MMR demonstrates excellent ability to retrieve relevant images because it considers multiple intrinsic structures of the images. We propose a novel cost function that is minimized to obtain the ranking scores of the images. Our proposed approach provides better results than the existing techniques. Furthermore, we present efficient techniques to create the multiple manifolds.

We remark that Huang *et al.* [6] also utilizes more than one low-level features. However, they construct only one manifold by using average manifold distance of multiple features. Since only a single manifold is used, the proposed approach does not preserve the original geometric structure of any of the features. In contrast, our approach constructs multiple manifolds and utilizes the geometric structure of each feature. This enables our approach to yield better results as demonstrated in our experiments. Furthermore, we show that our proposed approach is more efficient and can be used on large image databases.

Our contributions in this paper are summarized below.

- We propose multi-manifold ranking (MMR) that utilizes multiple intrinsic structures of the images to provide a better ranking of the images.

- To handle large image databases, we improve the efficiency of MMR by using singular value decomposition [1] as well as anchor graphs.
- Our extensive experimental results on real world image databases demonstrate that our algorithm provides better retrieval results than state of the art existing techniques (MR [4], ADF [6] and EMR [18]) that use a single manifold for image retrieval. Furthermore, the running time of our algorithm is similar to that of EMR and is significantly lower than those of MR and ADF. We also present simple extensions of MR and EMR that use more than one manifolds. Although these extended versions demonstrate better retrieval results than the original versions, our proposed multi-manifold ranking performs significantly better.

The rest of the paper is organized as follows. Related work is reviewed in Section 2. The details of multi-manifold ranking are presented in Section 3. Extensive experimental study on real world image databases is presented in Section 4. Section 5 concludes the paper.

2 Related Work

Zhou *et al.* [20] explored the importance of intrinsic geometrical structure of the data. They propose manifold ranking [21] that considers the intrinsic structure of the data for the ranking. Manifold ranking has been successfully used on various data types such as text [15], image [4] and video [19]. He *et al.* [4] are the first to use manifold ranking for image retrieval. While the proposed approach demonstrates good quality results, it is computationally expensive. Xu *et al.* [18] propose a more efficient approach that can efficiently handle large image databases. They replace the original image graph with anchor graph [8] which is significantly smaller in size but provides the results of similar quality. Huang *et al.* [6] use a probabilistic hypergraph for image retrieval. They construct a single manifold using the average manifold distance of multiple features.

All of the above manifold ranking based approaches consider geometric structure of a single image manifold which may not precisely represent the image content. Motivated by this, we propose a multi-manifold ranking based method for image retrieval which exploits the geometric structure of multiple manifolds each constructed using a different feature. Our idea for MMR is inspired by [3], which addresses the problem of video annotation through multi-graph using different video features.

3 Multi-Manifold Ranking

3.1 Preliminaries

Let X be a set containing n images, i.e. $X = \{x_1, x_2, \dots, x_n\}$. Multi-manifold ranking assigns each image x_i a ranking score F_i . $F = \{F_1, F_2, \dots, F_n\}$ is the ranking score vector containing the score of each image. $L = \{L_1, L_2, \dots, L_n\}$ is an indicator label vector where $L_i = 1$ if x_i is the query image, otherwise $L_i = 0$.

Multi-manifold ranking (MMR) constructs N graphs each using a different feature. G^k denotes a s -NN graph constructed on X using k^{th} feature. Specifically, G^k is constructed by connecting every two vertices x_i and x_j if one is

among the s nearest neighbors of the other. Here, the nearest neighbors are computed using Euclidean distance between the k^{th} feature vectors of the images. The Euclidean distance between the k^{th} feature vectors of x_i and x_j is denoted as $\|x_i, x_j\|_k$.

W^k denotes the edge affinity matrix of G^k . Each entry W_{ij}^k in W^k represents the similarity between x_i and x_j according to the k^{th} feature vector. W_{ij}^k is defined by a Gaussian kernel and is set to $\exp(-\|x_i, x_j\|_k^2/2\sigma^2)$ if there is an edge in G^k between x_i and x_j . Otherwise, W_{ij}^k is zero. D^k is the diagonal matrix of G^k where each element D_{ii}^k is defined as $D_{ii}^k = \sum_{j=1}^n W_{ij}^k$.

3.2 Objective Cost Function

In this section, we propose a novel cost function, inspired by [3], to obtain the ranking scores of the images in X . The cost function $O(F)$ considers N image manifolds each constructed using a different feature. The ranking score vector F is obtained by minimizing the cost function $O(F)$ given in Eq. 1.

$$O(F) = \frac{1}{2} \sum_{k=1}^N \left(\sum_{i,j=1}^n W_{ij}^k \left(\frac{1}{\sqrt{D_{ii}^k}} F_i - \frac{1}{\sqrt{D_{jj}^k}} F_j \right)^2 + \lambda \sum_{i=1}^n (F_i - L_i)^2 \right) \quad (1)$$

The first term ensures that nearby points (i.e., similar images in the multiple image manifolds) are assigned similar ranking scores. The second term is the fitting constraint which ensures that the ranking results should fit the initial label assignment. λ is the regularization trade-off parameter for the fitting constraint.

We minimize $O(F)$ by setting $\frac{\partial O(F)}{\partial F} = 0$, which leads to the following equation.

$$\sum_{k=1}^N \left((I - (D^k)^{-\frac{1}{2}} W^k (D^k)^{-\frac{1}{2}}) F + \lambda (F - L) \right) = \sum_{k=1}^N \left((1 + \lambda) F - S^k F - \lambda L \right) = 0 \quad (2)$$

where $S^k = (D^k)^{-\frac{1}{2}} W^k (D^k)^{-\frac{1}{2}}$. Note that Eq. 2 is equivalent to the following equation.

$$\sum_{k=1}^N \left(F - \frac{S^k}{1 + \lambda} F - \frac{\lambda}{1 + \lambda} L \right) = 0 \quad (3)$$

Let $\alpha = \frac{1}{1 + \lambda}$. Eq. 3 is equivalent to $\sum_{k=1}^N (I - \alpha S^k) F = N(1 - \alpha) L$. Hence, the final optimal ranking score vector denoted by F^* can be obtained as follows.

$$F^* = \left(\sum_{k=1}^N (I - \alpha S^k) \right)^{-1} N(1 - \alpha) L \quad (4)$$

where I is the identity matrix. Since both $(1 - \alpha)$ and N remain the same for all the images, they do not affect the retrieval results. Therefore, F^* can be obtained as follows.

$$F^* = \left(\sum_{k=1}^N (I - \alpha S^k) \right)^{-1} L \quad (5)$$

Eq. 5 is the closed form for the optimal solution F^* . In large scale problems, the *iteration scheme* is preferred [18]. Therefore, we also consider the iterative form which is given below.

$$F(t+1) = F(t) + \mu \sum_{k=1}^N (F(t) - S^k F(t) + \lambda(F(t) - L)) \quad (6)$$

where $F(t)$ is the ranking score vector at time stamp t . By setting $\mu = -\frac{1}{N(1+\lambda)}$, the following equation can be obtained.

$$F(t+1) = F(t) - \frac{1}{N(1+\lambda)} \sum_{k=1}^N ((1+\lambda)F(t) - S^k F(t) - \lambda L) = \frac{\sum_{k=1}^N (\alpha S^k F(t) + (1-\alpha)L)}{N} \quad (7)$$

Since N remains constant for all images, it is sufficient to consider the following equation which omits N .

$$F(t+1) = \sum_{k=1}^N (\alpha S^k F(t) + (1-\alpha)L) \quad (8)$$

The above iterative form can be used in the iterative scheme. During each iteration, each vertex (i.e., image) receives information from its neighbors (the first term) and retains its initial information (the second term). The iteration process is repeated until convergence. By following the arguments similar to [21], it can be shown that Eq. 8 is converged to the following equation when $F(0)$ is initialized to L .

$$F^* = \lim_{t \rightarrow \infty} F(t) = N(1-\alpha)(I - \alpha S)^{-1}L \quad (9)$$

Note that both N and $(1-\alpha)$ can be omitted from Eq. 9 without changing the final retrieval results because they are constant for all images. Therefore, the optimal ranking results can be obtained as follows.

$$F^* = \lim_{t \rightarrow \infty} F(t) = (I - \alpha S)^{-1}L \quad (10)$$

We remark that although Eq. 10 may assign negative scores to some of the images, the relative ranking order of the images is preserved. Nevertheless, if desired, the scores of all the images may be normalized (e.g., by shifting) such that each image gets a positive score.

3.3 Improving the Efficiency of MMR

The approach we mentioned in the previous section has two major limitations. Firstly, the time complexity for constructing the affinity matrix for n data points using s nearest neighbors is $\mathcal{O}(sn^2)$ [8]. Secondly, the inverse matrix computation in Eq. 5 requires $\mathcal{O}(n^3)$. Clearly, the cost of constructing the affinity matrix and inverse matrix computation is prohibitive for large image databases. Hence, this approach is not suitable for the large image databases.

The first limitation can be addressed by using anchor graphs [8] in a similar way as used in [18]. This reduces the cost from $\mathcal{O}(sn^2)$ to $\mathcal{O}(dmn)$ where $m \ll n$ and $d \ll n$. Next, we use singular decomposition to address the second limitation and reduce the cost from $\mathcal{O}(n^3)$ to $\mathcal{O}(m^3)$ where $m \ll n$.

Let I_r denote an identity matrix of size $r \times r$. Before we present the details of efficient matrix inversion, we prove that the following equation holds.

$$(NI_n - \alpha HH^T)^{-1} = \frac{I_n - H(H^T H - \frac{N}{\alpha} I_m)^{-1} H^T}{N} \quad (11)$$

Proof. We prove the correctness of the equation by showing that R.H.S. divided by L.H.S. equals to an identity matrix.

$$\begin{aligned} & \frac{(NI_n - \alpha HH^T) \left(\frac{I_n - H(H^T H - \frac{N}{\alpha} I_m)^{-1} H^T}{N} \right)}{NI_n - \alpha HH^T - (NH - \alpha HH^T H) \left(\frac{H^T H - (\frac{N}{\alpha} I_m)^{-1} H^T}{N} \right)} \\ &= \frac{NI_n - \alpha HH^T + \alpha H \left(-\frac{N}{\alpha} I_m + H^T H \right) \left(\frac{H^T H - \frac{N}{\alpha} I_m}{N} \right)^{-1} H^T}{NI_n - \alpha HH^T + \alpha HH^T} \\ &= \frac{NI_n - \alpha HH^T + \alpha HH^T}{NI_n - \alpha HH^T + \alpha HH^T} = \frac{NI_n}{NI_n} = I_n \end{aligned} \quad (12)$$

Based on Eq. 11, we show that the cost of the matrix operation can be reduced. Let H^k be defined as following.

$$H^k = (D^k)^{-\frac{1}{2}} Z^k (\Lambda^k)^{\frac{1}{2}} \quad (13)$$

The following equation can be verified.

$$H^k (H^k)^T = (D^k)^{-\frac{1}{2}} W^k (D^k)^{-\frac{1}{2}} \quad (14)$$

Recall that R.H.S. of Eq. 14 equals to S^k (see Eq. 2 in Section 3.2).

$$S^k = H^k (H^k)^T \quad (15)$$

We replace S^k in Eq. 5 with its value in Eq. 15 which yields the following.

$$F^* = (NI - \alpha \sum_{k=1}^N H^k (H^k)^T)^{-1} L \quad (16)$$

Note that each $H^k (H^k)^T$ is a symmetric matrix. Hence, $\sum_{k=1}^N H^k (H^k)^T$ is also a symmetric matrix. Without loss of generality, we set $S = \sum_{k=1}^N H^k (H^k)^T$ which is a $n \times n$ gram matrix.

$$F^* = (NI - \alpha S)^{-1} L \quad (17)$$

We decompose S by using singular value decomposition [1] $S = U \Lambda U^T$ such that $U^T U = I_n$. Note that the decomposition takes $\mathcal{O}(n^3)$ but it can be efficiently approximated in $\mathcal{O}(m^3)$ by using the techniques presented in [16]. Assume that we have obtained the approximate decomposition of S as follows.

$$S = U_m \Lambda_m U_m^T \quad (18)$$

where U_m is a $n \times m$ matrix formed by the first m normalized eigenvectors of U and m equals to the number of anchor images. A_m is the diagonal matrix with m diagonal elements (sorted in decreasing order from left to right) and correspond to the m largest eigenvalues of S . Eq. 18 is equivalent to the following equation.

$$S = U_m A_m^{\frac{1}{2}} A_m^{\frac{1}{2}} U_m^T = Y Y^T \quad (19)$$

where $Y = U_m A_m^{\frac{1}{2}}$. By combining Eq. 17 and Eq. 19, we obtain the following equation.

$$F^* = (N I_n - \alpha S)^{-1} L = (N I_n - \alpha Y Y^T)^{-1} L = \frac{I_n - Y(Y^T Y - \frac{N}{\alpha} I_m)^{-1} Y^T}{N} L \quad (20)$$

Since N remains constant for all of the images, the optimal ranking score vector F^* can be obtained as follows.

$$F^* = (I_n - Y(Y^T Y - \frac{N}{\alpha} I_m)^{-1} Y^T) L \quad (21)$$

Note that Eq. 21, requires the inversion of a $m \times m$ matrix in contrast to Eq. 5 that requires the inversion of a $n \times n$ matrix. Hence, Eq. 21 reduces the cost from $\mathcal{O}(n^3)$ to $\mathcal{O}(m^3)$.

4 Experimental Results

In this section, we evaluate the performance of our proposed approach (MMR) by using several real world image databases. All the experiments are implemented in Matlab R2009a and C++. First, we present the experimental setup in Section 4.1. Then, in Section 4.2, we evaluate the performance of our proposed approach.

4.1 Experimental Setup

Data sets. We evaluated the performance of MMR on the following data sets.

- **COREL:** It is composed of 7700 images divided into 77 categories.
- **Caltech101:** This image database contains 8677 images from 101 different categories.
- **MSRC:** The data set contains 18 different categories and consists of approximately 4300 images.

Competitors. We compare our proposed approach with several manifold ranking based algorithms. Below are the details.

- **MR.** This is the first work [4] that applied manifold ranking (MR) for image retrieval.
- **EMR.** This is the algorithm proposed by Xu *et al.* [18]. While MR demonstrated good quality results, it is not suitable for large scale image databases because of its high computational cost. EMR proposes interesting techniques to improve the efficiency of MR and demonstrates that it retrieves the results of similar quality.

•**ADF**. This algorithm is proposed in [6]. ADF uses multiple features to construct a single image manifold.

Recall that our proposed approach uses multiple features to construct multiple image manifolds. We argue that using multiple image manifolds yield better results than the previous techniques. A natural question is whether previous approaches (e.g., MR and EMR) can perform better if they also utilize more than one features. To answer this question, we extend the previous techniques such that they utilize multiple features. Below are the details of how each technique is extended.

•**MR**^{+N}. N denotes the total number of features used by the algorithm. Let F_i^k be the ranking score of x_i computed by MR [4] using k^{th} feature. The final score of each image x_i is $\sum_{k=1}^N F_i^k$. MR^{+N} ranks the images according to the final scores. Note that MR⁺¹ is the same as original MR algorithm proposed in [4].

•**EMR**^{+N}. Similar to MR^{+N}, EMR^{+N} computes the score of each image according to each feature. The images are then ranked according to their final scores. We remark that EMR⁺¹ is the original EMR algorithm proposed in [18].

Later, we show that these extended versions retrieve better results than their respective original versions. Moreover, the quality of the retrieved results improves as the value of N increases. Similar to the notations used for extended version of MR and EMR, we use MMR^{+N} to denote that our algorithm MMR was run using N features. Similarly, ADF^{+N} denotes that ADF was run using N features.

Features used by the algorithms. We use some of the most popular features in the algorithms. More specifically, we use DoG-SIFT (Scale-Invariant Feature Transform) [9], HOG (Histogram of Oriented Gradients) [2], LBP (Local Binary Patterns) [11], Centrist [17] and RBG-SIFT [14]. Table 1 shows these features in a particular order. Any algorithm using N features uses the first N features shown in Table 1. For example, MMR⁺³ is our algorithm and uses the first three features (DoG-SIFT, HOG and LBP). Similarly, EMR⁺² denotes that EMR was run using first two features (DoG-SIFT and HOG). ADF⁺⁵ denotes that ADF was run using all of the features. We remark that this order of the features best suits EMR^{+N} which is our main competitor.

Table 1. Features used by the algorithms

	1	2	3	4	5
Feature	DoG-SIFT	HOG	LBP	Centrist	RGB-SIFT

Evaluation metric. Each image in the image databases has its own category label (e.g., car, aeroplane etc.). A query is randomly selected from these databases and a retrieved result is considered correct if its label matches with the query label. For each query, we retrieve top- K results where the default value of K is 10 unless mentioned otherwise. We use precision as the main evaluation metric which corresponds to the number of correct results in the top- K retrieved results divided by K . Since K is fixed for all competitors, the recall value is directly related to the precision, i.e., if precision is high then the recall is also high and vice versa. Hence, we use precision as the only evaluation metric.

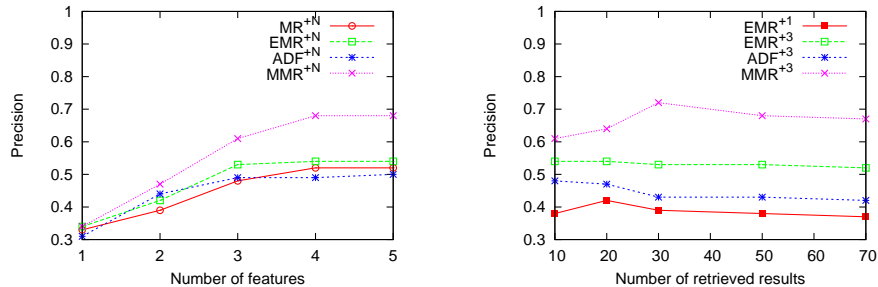
Recall that our algorithm (MMR) samples m anchor points and, for each image x_i , x_i is connected to its d nearest neighbors. We set m to 500 and d to 5

because our preliminary experimental evaluation demonstrated that these values of m and d give a reasonable trade-off between the precision and efficiency of the algorithm.

4.2 Performance comparison

In this section, we compare the performance (precision and efficiency) of our algorithm with the other competitors. At the end, we present a case study where we show the top-10 results returned by MMR, EMR and ADF for three queries.

Precision. In Fig. 2, we increase the number of features used by each algorithm and study its affect on the precision. Note that the performance of each algorithm improves as it uses more features. However, the precision obtained by our algorithm (MMR^{+N}) is the highest. This is because our algorithm constructs multiple manifolds and minimizes the cost function to obtain the ranking scores in contrast to the other algorithms that use multiple features (manifolds) somewhat trivially. Note that the improvement in precision is less significant when $N > 3$. Since the running time increases with the increase in N , we choose $N = 3$ for rest of the experiments (unless mentioned otherwise).



(a) Effect of number of features on precision

(b) Effect of K on precision

Fig. 2. Effect of number of features and K on precision.

As noted in [18] and observed from Fig. 2 (a), the precision of EMR^{+N} and MR^{+N} is quite similar. Furthermore, EMR^{+N} is more efficient than MR^{+N} as we demonstrate later. Therefore, for a clearer illustration of results, in the rest of the experiments we exclude MR^{+N}. In Fig. 2 (b), we issue top- K queries and vary K from 10 to 70 and study its affect on the precision. We observe that the precision of each of the algorithms remain unaffected with the increase in K . Also, note that our algorithm consistently gives better results than the other competitors.

In Fig. 3, we study the precision at a more detailed level. More specifically, we randomly choose 90 categories from the three image databases. For each category, we randomly choose one image as the query. For each query, we obtain top-10 results and record the precision. Fig. 3 shows the precision of each algorithm for the queries selected from each of the 90 categories. It can be observed that our approach MMR⁺³ consistently performs better than the other methods. The EMR proposed in [18] (shown as EMR⁺¹) has the lowest precision. However, EMR⁺³ that uses three manifolds has better retrieval performance than ADF^{+N}.

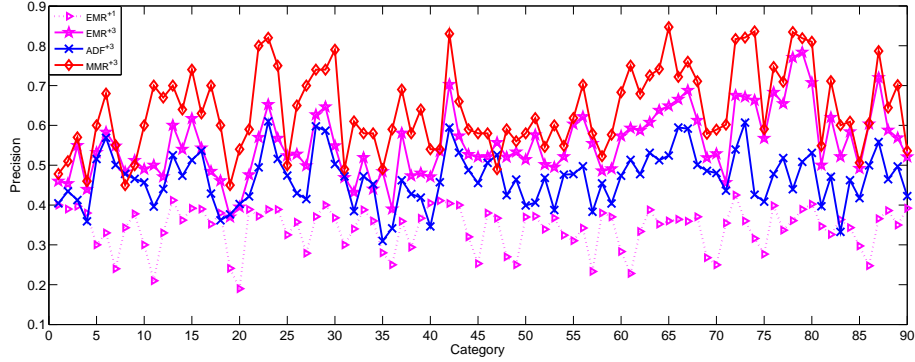


Fig. 3. Precision of top-10 images for randomly selected queries from each category

Running time. In Fig. 4(a), we increase the number of features used by each algorithm and study its affect on the running time. Note that the running times of ADF^{+N} and MR^{+N} are much higher than the running time of our algorithm (MMR^{+N}) and EMR^{+N} . This is because MMR^{+N} and EMR^{+N} present efficient techniques for matrix inversion and use the anchor graphs to approximate the large image graphs. Also, note that EMR^{+N} and MMR^{+N} scale better as the number of features increases. The cost of MR^{+N} is the highest. In order to better illustrate the performance of other approaches, we do not display the cost of MR^{+N} when $N > 2$.

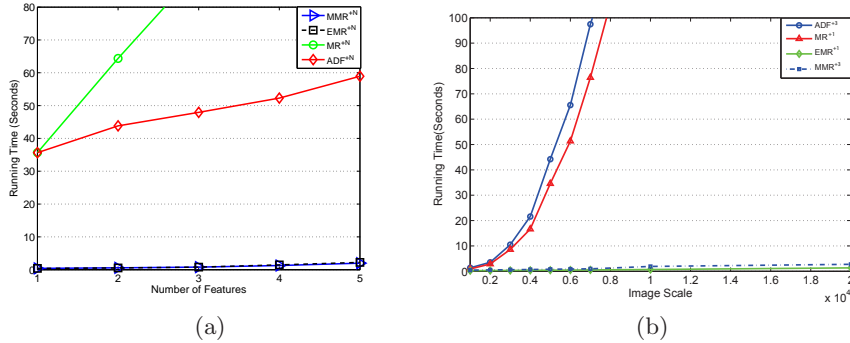


Fig. 4. Effect of number of features and image database size on running time

In Fig. 4(b), we increase the size of image databases and study its affect on the running times of all algorithms. It can be observed that ADF^{+N} and MR^{+N} cannot handle large scale databases (e.g., the running time is more than 80 seconds when the image database contains 8000 images). On the other hand, our proposed algorithm scales better and can handle large scale image databases. The cost of EMR^{+1} is the lowest. This is because it uses a single image manifold whereas our algorithm MMR^{+3} uses three image manifolds. Nevertheless, the running times of both of the algorithms are quite close to each other.



Fig. 5. Three queries are issued and top-10 results returned by MMR^{+3} , ADF^{+3} and EMR^{+3} are displayed. The irrelevant images retrieved by our algorithm are marked with red square. It can be noted that MMR^{+3} returns more relevant results than the other two algorithms.

A case study. In this section, we display the top-10 results returned by MMR^{+3} , ADF^{+3} and EMR^{+3} for three different queries. Fig. 5 displays the results returned by each of the algorithms. Irrelevant results returned by our algorithm are denoted by red square. Note that our algorithm returns more relevant results than the other two algorithms.

5 Conclusion

In this paper, we propose a novel method name multi-manifold ranking (MMR) which uses multiple image manifolds for image retrieval. We conduct extensive experimental study on real world image databases and demonstrate that MMR provides better retrieval results than state of the art techniques. Our experimental results demonstrate that our algorithm is much more efficient than two existing algorithms and is comparable to the most efficient existing approach.

Acknowledgments: Muhammad Aamir Cheema is supported by ARCDP130103405 and ARCDE130101002; Xuemin Lin is supported by ARC DP0987557, ARC DP110102937, ARC DP120104168 and NSFC61021004.

References

1. P. R. Christopher D. Manning and H. Schtze. *An introduction to information Retrieval*. Cambridge University Press, 2009.

2. N. Dalal and B. Triggs. Histogram of oriented gradients for human detection. In *CVPR*, pages 886–893, 2005.
3. M. W. et al. Unified video annotation via multigraph learning. *IEEE Trans. Circuits Syst. Video Techn.*, 19(5):733–746, 2009.
4. J. He, M. Li, H. Zhang, H. Tong, and C. Zhang. Manifold-ranking based image retrieval. In *ACM Multimedia*, pages 9–16, 2004.
5. J. Huang, S. R. Kumar, M. Mitra, and W. jing Zhu. Spatial color indexing and applications. *International Journal of Computer Vision*, 35(3):245–268, 1999.
6. Y. Huang, Q. Liu, S. Zhang, and D. N. Metaxas. Image retrieval via probabilistic hypergraph ranking. In *CVPR*, pages 3376–3383, 2010.
7. D. Liu, X.-S. Hua, L. Yang, M. Wang, and H.-J. Zhang. Tag ranking. In *ACM WWW*, pages 351–360, 2009.
8. W. Liu, J. He, and S.-F. Chang. Large graph construction for scalable semi-supervised learning. In *ICML*, pages 679–686, 2010.
9. D. G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004.
10. B. S. Manjunath, J. rainer Ohm, V. V. Vasudevan, and A. Yamada. Color and texture descriptors. *IEEE Transactions on Circuits and Systems for Video Technology*, 11:703–715, 1998.
11. T. Ojala, M. Pietikainen, and T. Maenpaa. Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(7):971–987, 2002.
12. M. J. Swain and D. H. Ballard. Color indexing. *International Journal of Computer Vision*, 7(1):11–32, 1991.
13. A. Ulges, M. Worring, and T. M. Breuel. Learning visual contexts for image annotation from flickr groups. *IEEE Transactions on Multimedia*, 13(2):330–341, 2011.
14. K. E. A. van de Sande, T. Gevers, and C. G. M. Snoek. Evaluating color descriptors for object and scene recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(9):1582–1596, 2010.
15. X. Wan, J. Yang, and J. Xiao. Manifold-ranking based topic-focused multi-document summarization. In *IJCAI*, pages 2903–2908, 2007.
16. C. K. I. Williams and M. Seeger. Using the nystrm method to speed up kernel machines. In *NIPS*, pages 682–688, 2000.
17. J. Wu and J. M. Rehg. Centrist: A visual descriptor for scene categorization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(8):1489–1501, 2011.
18. B. Xu, J. Bu, C. Chen, D. Cai, X. He, W. Liu, and J. Luo. Efficient manifold ranking for image retrieval. In *ACM SIGIR*, pages 525–534, 2011.
19. X. Yuan, X.-S. Hua, M. Wang, and X. Wu. Manifold-ranking based video concept detection on large database and feature pool. In *ACM Multimedia*, pages 623–626, 2006.
20. D. Zhou, O. Bousquet, T. N. Lal, J. Weston, and B. Schlkopf. Learning with local and global consistency. In *NIPS*, pages 592–602, 2003.
21. D. Zhou, J. Weston, A. Gretton, O. Bousquet, and B. Schlkopf. Ranking on data manifolds. In *NIPS*, 2003.