Refining Content Based Image Retrieval via Semi-Supervised Learning

Songhao Zhu and Yuncai Liu

Institute of Image Processing and Pattern Recognition, Shanghai Jiao Tong University
800, Dong Chuan Road, 200240, Shanghai China
play_tree@163.com, whomliu@sjtu.edu.cn

ABSTRACT

Content-based image retrieval plays a key role in the management of a large image database. However, the results of existing approaches are not as satisfactory for the gap between visual features and semantic concepts. Therefore, a novel scheme is here proposed. First, to tackle the problem of large computational cost involved in a large image database, a pre-filtering processing is utilized to filter out the most irrelevant images while keeping the most relevant ones. Second, the relevance between the query image and the remaining images is measured and the obtained relevance scores are stored for a later refinement processing. Finally, a semi-supervised learning algorithm is utilized to refine candidate ranking by taking into account both the pairwise information of unlabeled images and the relevance scores between the input query image and unlabeled images. Experiments conducted on a typical Corel dataset demonstrate the effectiveness of the proposed scheme.

1. INTRODUCTION

With the popularity of digital cameras, the amount of digital images available on personal devices has increased exponentially, which has brought about great challenges for designing efficient tools to retrieve a group of images of interest from a large image database. The key issue to construct a successful content-based image retrieval system is how to establish and represent the perceptual similarity between the huge volume of available information and a given image.

A common problem faced by researchers in real applications is: only a small number of positive images are given, and then a model needs to be learned to achieve semantically similar images for a query image. Striving to overcome the limitation of the lack of negative samples, many researchers have proposed different solutions. In [1], this issue is tackled by treating several unlabeled samples as the pseudo-negative samples for Support Vector Machine, a discriminative model. He *et al.* in [2] handle this problem by the manifold-ranking algorithm, which can propagate labels from positive training samples to testing samples. In [3], feedback from users is incorporated into to the system to form an iterative refinement processing. Furthermore, many other approaches are presented to deal with the issue of content-based image retrieval [4-8]. In spite of encouraging performance has been obtained, the content-based image retrieval problem has not been satisfactorily solved due to the large gap between low-level features and high-level semantic concepts, i.e., images of dissimilar semantic content may share some common low-level features, while images of similar semantic content may be scattered in the feature space.

In this paper, we propose a novel learning framework named random walk with restart based image retrieval (RWRIR), which is inspired by a recently developed random walk with restart algorithm (RWR) [9-11]. In RWRIR, relevance between the query image and database images is measured by projecting them to data points in the feature space and researching the relationship between them, which can effectively address the limitation of similarity metrics based on pairwise distance. Compared with those pairwise metrics, the relevance score defined by RWR can capture the global structure of the graph; compared with those traditional graph distances (such as shortest path, maximum flow etc), it can capture the multi-facet relationship between two nodes.

Motivated by the work in [9-11], we propose the framework as shown in Figure 1. First, when a large-scale image database is incorporated, the computation cost will be expensive. A possible solution to this problem is to introduce a pre-filtering processing, which can filter out the majority of irrelevant images whilst retaining the most relevant ones in accordance with the results of the manifold-ranking algorithm. Then, a probability density estimation algorithm is adopted to obtain the relevance scores between the input query image and the remained images in the database, and the relevance scores are stored as the restart vector for later retrieval processing. And last, to fully make use of both the pair-wise information of unlabeled images and the relevance scores between input query image and unlabeled images, image-based semi-supervised learning, random walk with restart algorithm is exploited to achieve the images perceptually similar to the given query image.

The organization of this paper is as follows. Section 2 details the pair-wise similarity of images and the pre-filtering processing. Section 3 and Section 4 introduce the method of computing the relevance scores of samples and the retrieval processing using the random walk with restart algorithm respectively. Experimental results are provided in Section 5, followed by the concluding remarks in Section 6.

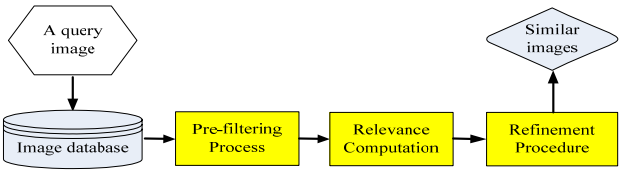

Figure 1. Proposed content-based image retrieval scheme.

2. CALCULATING PAIRWISE SIMILARITY AND PROFITEERING PROCESSING

In this section, we first introduce the pairwise similarity of samples by taking into account the influence of both distance and neighbourhood. Then, we analyze the importance and necessity of the pre-filtering processing.

2.1 Pairwise Similarity of Images

According to [12], there are two typical assumptions used in the graph-based semi-supervised learning method. The first assumption is the neighbourhood assumption: nearby samples are likely to have the same label. The second assumption is the structure assumption: samples on the same "structure" (typically referred to as a cluster or a manifold) are likely to have the same label. Therefore, the similarity estimation between images plays a crucial role for graph-based methods as it is the basis of label propagation. Based on the first assumption and the experimental results in [13], the similarity between images i and j is here estimated using the metric of Manhattan distance:

$$S_d(i,j) = \exp\left[-\frac{|x_i - x_j|}{\sigma}\right] = \prod_{k=1}^{K} \exp\left[-\frac{|x_i^k - x_j^k|}{\sigma^k}\right] \quad (1)$$

where x_i^l is the k^{th} dimension in the low-level feature vector x_i of sample i, K is the dimension of the feature space, and σ^k is the positive parameter that reflects the scope of different dimension. According to the second assumption and for the purpose of consistency, the similarity between samples i and j is also formulated based on the Manhattan distance:

$$S_n(i,j) = \exp\left(-\frac{|n_i - n_j|}{\sigma}\right) \quad (2)$$

where n_i is the estimation of neighbourhood probability density of sample i, and n_i is defined as below:

$$n_i = \frac{1}{N_i}\sum_{p=1}^{N_i}\prod_{k=1}^{K}\exp\left[-\frac{|x_i^k - x_p^k|}{\sigma^l}\right] \quad (3)$$

where N_i is the set of neighbours of sample i.

Taking into account the first assumption and the second assumption, the pairwise similarity between samples i and j is formulated as:

$$\begin{aligned} S(i,j) &= S_d(i,j) \bullet S_n(i,j) \\ &= \exp\left[-\frac{|x_i - x_j|}{\sigma}\right] \bullet \exp\left(-\frac{|n_i - n_j|}{\sigma}\right) \end{aligned} \quad (4)$$

where '•' denotes the Hadamard product, and $S_d(i,j)$ is the normal similarity matrix. The first term on the right side of (4) shows that the similarity between two samples decreases with respect to the increment of their distance in the feature space; and the second one indicates that the similarity decreases with the increment of the density difference.

2.2. Profiteering Processing

As aforementioned, the computational cost renders many algorithms intractable when facing large datasets. To tackle this problem, we make use of an efficient pre-filtering process to filter out the majority of most irrelevant unlabeled samples whilst preserving the most relevant ones.

After analyzing the entire image retrieval processing, it can be seen that the pre-filtering processing should simultaneously satisfy two criteria: one is the low computational cost and the other is the high recall rate. Here a modified nearest neighbour rule is utilized to implement the pre-filtering processing. More specifically, for a query sample, samples in the database are ranked with respect to the pairwise similarity obtained using Equation (4): the larger the value of pairwise similarity for a sample, the higher the ranking score it has. After ranking images according to the pairwise similarity value, a specified percentage of samples are filtered out. In this situation, the whole computational cost can scale from $O(M^3)$ (here M contains both labelled and unlabeled images) to $O(N^3)$ (here N is the number of retained images for the query image and $N << M$), therefore the computational cost is significantly reduced.

The reason for this pre-filtering processing is to preserve enough connection of relevant samples while preserving the sparse property of the weighted graph for later retrieval processing. Namely, the constructed graph in this way is a disconnected graph, and an inevitable consequence of a disconnected graph is that not all the images will end up with a ranking score. According to the analysis, we can note that the images without ranking scores are irrelevant ones of the query image. Thus we can conclude with high confidence that omitting the images with no ranking score will not influence the overall performance.

3. RELEVANCE SCORES

After a query image is given, a scale should be introduced to describe the correlation between the query image and the retained images in the database. This scale is also called the fuzziness, which stands for the confidence measure of one image perceptually similar to the given query image, and is represented by the fuzzy numbers with range of [0, 1]. In our situation, each retained image is assigned a real-value to indicate how confidently it belongs to the same categorization as the query image. Several functions are often used to convert the distance values into fuzzy membership values [14]. To compute the membership values, we use the hyperbolic tangent function. The fuzzy function takes the distance between an image in the testing set and the query image as the input and maps it to a value in the range of [0; 1].

$$q(i) = \frac{\tanh\left(\frac{|x_q - x_i| + s_c}{d_c}\right) - \tanh\left(\frac{|x_q - x_i| - s_c}{d_c}\right)}{2 \times \tanh\left(\frac{s_c}{d_c}\right)} \quad (5)$$

Given a query image q with the feature vector x_q, the membership, $q(i)$, of a image i in the testing set with the feature vector x_i belonging to the same categorization is computed using Equation (5), where $tanh(.)$ is the Hyperbolic tangent function, d_c is average distance between the centres of different image clusters, and s_c is the spread of the function which controls the shape. The average distance between R different image cluster centres c_r is computed as below:

$$d_c = \frac{2}{R(R-1)} \sum_{r=1}^{R} \sum_{t=r+1}^{R} | c_r - c_t | \quad (6)$$

The shape of the function can be more concentrated to the function mean when s_c decreases. Correspondingly, if the user prefers a lose condition, s_c can be set larger, which ensures that the feature relatively far from the cluster centre can still have high membership value.

For each retained image in the database, its membership value to the query image is computed and the image retrieval is later performed based on the membership plot. One example of the fuzzy membership plot is shown in Figure 2.

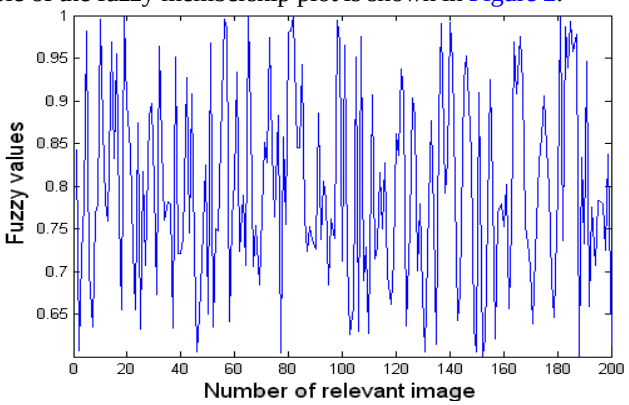

Figure 2: Fuzzy memberships of an example query image. For each query image, the number of relevant images is chosen as 200 after pre-filtering processing.

4. THE BASIC ALGORITHM

Once we achieve a set of membership values for a query image, we can use it to perform the task of content-based image retrieval. Here, the retrieval process can be considered as a ranking process of the retained images according to the nearest neighbourhood rule. In order to consider both the membership values of the query image and the correlations between images, we adopt the idea of similarity propagation via graph learning in the ranking process and solve the content-based image retrieval problem with an improved random walk with restart algorithm [11].

4.1. Constructing Similarity Graphs

The process of constructing a similarity graph G of a query image q contains following two main steps: vertexes setting and vertexes connecting.

Step 1: Vertexes Setting. For a query image q, let $A=\{1, 2,..., i,..., N\}$ be a set of N images generated by the nearest neighbourhood. Then, each image i can be regarded as a vertex of the similarity graph G. The form of the similarity graph G is $G=(V,E)$, where V is the image set $A=\{1, 2,..., i,..., N\}$ of the query image q, and all vertexes are connected with weighted edges set E.

Step 2: Vertexes Connecting. The edges set E of the similarity graph G are weighted by the $N*N$ affinity matrix W with entry $w_{i,i}=0$ and $w_{i,j}$ when $i \neq j$ as shown in Equation (7):

$$w_{i,j} = S(i,j) = S_d(i,j) \bullet S_n(i,j) \quad (7)$$

4.2. Refining Candidate Ranking

The image retrieval procedure consists of following several phases: setting restart vector, constructing normalized weighted matrix and performing refining processing.

Step 1: Setting Restart Vector. In this work, membership values of the query image q, $qm=\{q(1), q(2), ..., q(N)\}$ are chosen as the restart vector e, and e is normalized to make sure that the sum of all components in e is one.

Step 2: Constructing Normalized Weighted Matrix. The affinity matrix W calculated in sub-section 4.1 is chosen as the weighted matrix W in the random walk with restart procedure. Furthermore, each column in W is also normalized to ensure that the sum of each column in W is one.

Step 3: Running Refining Processing [11].

Step 3.1: Partitioning the similarity graph G into k partitions via the partition algorithm in [15].

Step 3. 2: Decomposing the normalized weighted matrix W into two diagonal matrices: $W=W_1+W_2$, according to the partition result obtained in step 3.1, where the diagonal matrix W_1 contains all within-partition links and W_2 contains all cross-partition links.

Step 3.3: Using Equation (10) to denote the components of the diagonal matrix W_1:

$$W_1 = \begin{pmatrix} W_{1,1} & 0 & \dots & 0 \\ \dots & \dots & \dots & \dots \\ \dots & \dots & 0 & W_{1,N} \end{pmatrix} \quad (10)$$

Step 3.4: Computing each corresponding component $\sigma_{1,j}^{-1}$ for each component $W_{1,j}$ of the diagonal matrix W_1:

$$\sigma_{1,j}^{-1} = \left(I - \alpha * W_{1,j}\right)^{-1} \quad (11)$$

where α is a parameter that control the probability of returning to the vertex j. Furthermore, the

Step 3.5: Using Equation (12) to perform the low-rank approximation for W_2 to achieve a vertex-concept matrix U and a concept-concept matrix S:

$$W_2 = USU^T \quad (12)$$

where each column of U is the eigenvector of W_2 and S is a diagonal matrix, whose diagonal elements are eigenvalues of W_2. In addition, the superscript 'T' denotes transpose.

Step 3.6: Achieving final membership values of the query image q using following Equation:

$$\gamma = (1-\alpha) * (\sigma_1^{-1} + \alpha * \sigma_1^{-1} * U * \Lambda * U^T * \sigma_1^{-1}) * e \quad (13)$$

where

$$\Lambda = (S^{-1} - \alpha * U^T * \sigma_1^{-1} * U)^{-1} \quad (14)$$

The i^{th} element of $N*1$ vector γ is the final membership value for image i, and top V images with the highest probability are selected as the retrieved images for each query image q. This way of selecting final retrieved images is also named 'top @ V' in other literatures.

5. EXPERIMENTAL RESULTS

5.1. Experiment Design

To test the proposed method and compare it with other methods described in the literature, a general purpose image database, the 5000 Corel dataset are used as the ground truth database. This dataset is manually subdivided into 50 semantic classes, such as building, flower, horse, tiger, beach, building, and sunset, and so on. Each class contains 100 images on the

same topic. Each image in the whole database is used as a query, and the average of the results over all the 5,000 images is used to evaluate the system performance.

Obviously feature selection is a large open problem and has a great influence on the performance of image retrieval. In the current implementation, the low-level feature vector used here consists of the features of colour histogram, colour moment and wavelet texture:

- 128-D colour histogram in HSV colour space with 8 bins for Hue and 4 bins each for Saturation and Value;
- 225-D block-wise colour moments in LAB colour space, which are extracted over 5X5 fixed grid partitions, each block is described by a 9-D feature;
- 36-D Pyramid Wavelet texture, which is extracted over 6-level Haar Wavelet transformation, each level is described by a 6-D feature: mean and variance of coefficients in high/high, high/low, and low/high bands.

There are three parameters needed to be set in the proposed algorithm: N, l_c and α. The values of l_c and α are decided through cross validation. In our experiments, we set the number of neighbourhood N as 200, and the values of l_c and α are 1.2 and 0.4 respectively.

5.2 Experimental Results

To test the performance of the proposed scheme, each image is chosen as a query. Since each class contains 100 images, there should be at most 100 possible matches to a query image. Here the mean of the weighted precision and the average rank are adopted to evaluate the performance of various methods. The retrieval precision of a query image q is:

$$p_l(q) = \frac{|l(q) \cap n_l(q)|}{|l(q)|} \tag{15}$$

where $l(q)$ is the set of top l retrieved images, and $n_l(q)$ is the set of relevant images for a given query image q. Based on Equation (15), the weighted precision of a query image q is:

$$p(q) = \frac{1}{100}\sum_{l=1}^{100} p_l(q) = \frac{1}{100}\sum_{l=1}^{100} \frac{|l(q) \cap n_l(q)|}{|l(q)|} \tag{16}$$

The formulation of the mean of the weighted precision with each class C_k, $1 \le k \le 50$, is as below:

$$p_k = \frac{1}{100}\sum_{q \in C_k} p(q) \tag{17}$$

Rank is referred to as the position of relevant images in the returned results, which is denoted ay $p(n_l)$. Average rank $r(q)$ is the mean of ranks of all the correct matches for q:

$$r(q) = \frac{1}{100}\sum_{l=1}^{100}\sum_{n_l=1}^{l} p(n_l) \tag{18}$$

Similarly, the mean values of the average rank within each class C_k, $1 \le k \le 50$, is as below:

$$r_k = \frac{1}{100}\sum_{q \in C_k} r(q) \tag{19}$$

We compare retrieval results using the proposed scheme with those obtained with Support Vector Machine method (SVMM) [1] and Manifold-Ranking method (MRM) [2]. The comparison of the average of the weighted precision (p_k) and mean rank (r_k) of ten example categories are shown in Figure 3, respectively.

Retrieval performance can be further quantified as follows. For a query image q and a positive integer $l(q)$, $n_l(q)$ is the number of relevant images among the top $l(q)$ retrieval. $p_l(q)$ and $r_l(q)$ are precision and recall rates for top $l(q)$ retrieval for a query image q respectively:

$$p_l(q) = \frac{|l(q) \cap n_l(q)|}{|l(q)|} \quad \text{and} \quad r_l(q) = \frac{|l(q) \cap n_l(q)|}{100} \tag{20}$$

The average precision and average recall over all the database images for the top $l(q)$ retrieval are::

$$p_l = \frac{1}{5000}\sum_{q=1}^{5000} p_l(q) \quad and \quad r_l(q) = \frac{1}{5000}\sum_{q=1}^{5000} r_l(q) \tag{21}$$

Equation (23) is used to denote the average precision and average recall within each category:

$$p_l = \frac{1}{100}\sum_{q \in C_k} p_l(q) \quad and \quad r_l(q) = \frac{1}{100}\sum_{q \in C_k} r_l(q) \tag{22}$$

Figure 4 shows values of p_l and r_l under different approaches, i.e., the proposed approach, the SVMM [1] and MRM [2].

As shown in Figure 3 and 4, our proposed method achieves the best performance than manifold-ranking based model and SVM based model. That is, the random walk with restart based model improves the performance of retrieval. This improvement can be analyzed from two aspects. One is that the pairwise similarity based on the neighbourhood assumption cannot sufficiently reflect the contribution form one image to another in the manifold-ranking based model, while our method implements the 'anisotropic' contribution by simultaneously taking into account the two assumptions. Another is that considering both the pairwise information of images in the testing set and the relevance scores between input query image and testing images also helps to enhance the retrieval accuracy.

6. CONCLUSIONS

This paper has presented a novel scheme to implement the issue of image retrieval. To consider both the membership values of the query image and the correlations between the query image and testing images, the random walk with restart algorithm is here utilized to solve a difficult case that only a small number of positive images are available. To decrease the computational cost, a pre-filtering processing has been introduced into the proposed scheme. Then, the relevant scores for each query image are represented by membership values. Experimental results have demonstrated the effectiveness of the proposed scheme.

Since feature selection is a large open issue and has great impact on the experimental results, our feature work will introduce other features into the scheme to further improve system performance. Our further work also includes conducting experiments on other large database. Furthermore,

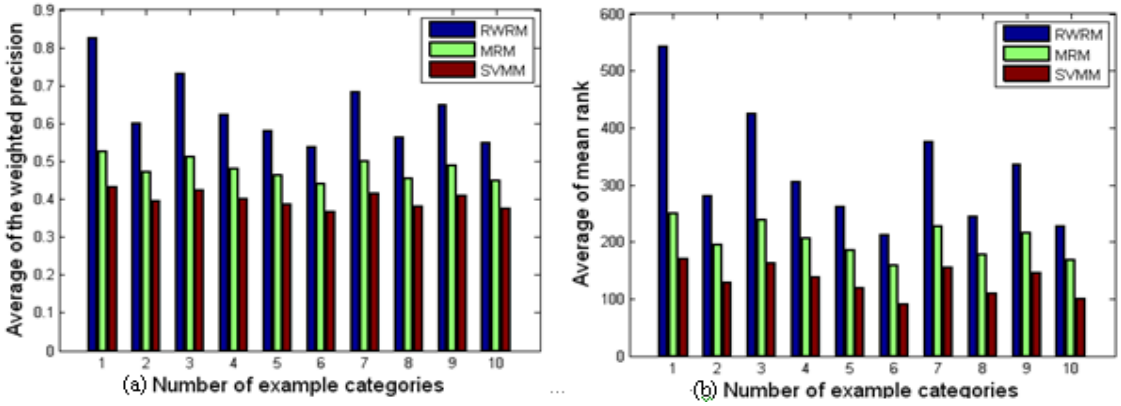

Figure 3. Comparison of (a) average of the weighted precision and (b) average rank within each example class.

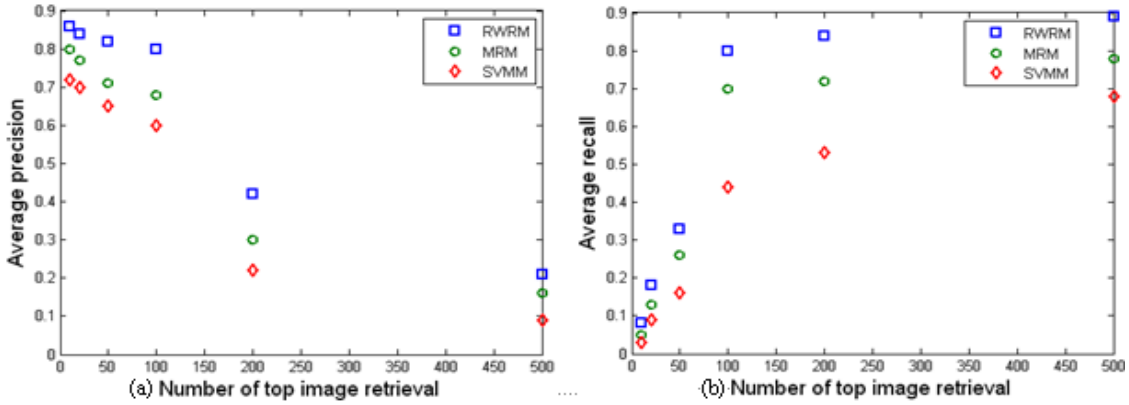

Figure 4. Comparison of (a): Average precision and (b): Average recall.

the relevance feedback is another research direction in our further work.

7. ACKNOWLEDGEMENTS

This work is supported by the National Key Basic Research and development Plan of China (973) under Grant No. 2006CB303103, and also supported by National High Technology Research and development Plan of China (863) under Grant No. 2009AA01Z330 and the National Natural Science Foundation of China under Grant No. 60833009.

8. REFERENCES

[1] Natsev, A., Naphade, M.R. and Tesic, J., "Learning the Semantics of Multimedia Queries and Concepts from a Small Number of Examples," in proc. ACM Conf. on Multimedia 2005, pp: 598 – 607.

[2] He, J.R., Li, M.J., Zhang, H.J., Tong, H.H. and Zhang, C.S., "Manifold-Ranking Based Image Retrieval," in proc. of ACM Conf. on Multimedia 2004, pp:9-16.

[3] Cui, J.Y. and Zhang, C. S., "Combining stroke-based and selection-based relevance feedback for content-based image retrieval," in proc. of ACM Conf. on Multimedia 2007, pp: 329-332.

[4] Giang, P., Worring, M. and Arnold W. M., "Similarity Learning via Dissimilarity Space in CBIR," in proc. of ACM Workshop on Multimedia Information Retrieval 2006, pp: 107-115.

[5] Shen, J. and Shepherd, J., "Efficient benchmarking of content-based image retrieval via resampling," in proc. of ACM Conf. on Multimedia 2006, pp: 569-578.

[6] Pham, T. T., Maillot, N. E., Lim, J. H., and Chevallet, J. P., "Latent semantic fusion model for image retrieval and annotation," in proc. of ACM Conf. on information and knowledge management 2007, pp: 439-443.

[7] Chang, H. and Yeung, D. Y., "Locally smooth metric learning with application to image retrieval," in proc. Of Int. Conf. on Computer Vision 2007, pp: 1-7.

[8] Pramod, S. K. and Jawahar, C. V., "Probabilistic Reverse Annotation for Large Scale Image Retrieval," in proc. Of Computer Vision and Pattern Recognition 2007, pp: 1-6.

[9] Zhou, D., Bousquet, O., Lal, T. N., Weston, J. and Scholkopf, B., "Learning with local and global consistency," in proc. Of Neural Information Processing Systems 2003, pp: 321--328.

[10] Pan, J.-Y., Yang, H.-J., Faloutsos, C. and Duygulu, P., "Automatic multimedia cross-modal correlation discovery," in proc. Of Knowledge Discovery and Data Mining 2004, pp: 653–658.

[11] Tong, H. H., Faloutsos, C. and Pan, J. Y., "Fast Random Walk with Restart and Its Applications," in proc. Of Knowledge Discovery and Data Mining 2006, pp: 613-622.

[12] Zhou D. Y., Bousquet O., Thomas N. L., Jason W., Bernhard S. O., "Learning with Local and Global Consistency," in Proc. Neural Information Processing Systems 2003, pp: 321-328.

[13] Kokare, M., Chatterji, B.N., and Biswas, P.K., "Comparison of similarity metrics for texture image retrieval," in Proc. Convergent Technologies for Asia-Pacific Region 2003, pp: 571-575.

[14] Mitaim, S., and Kosko, B., "The Shape of Fuzzy Sets in Adaptive Function Approximation," IEEE Transactions on Fuzzy Systems, Vol. 9, No. 4, pp: 637-656, 2001.

[15] Karypis G. and Kumar V., "Parallel multilevel k-way partitioning for irregular graphs," SIAM Review, Vol. 41, No. 2, pp: 278–300, 1999.