An efficient Re-ranking of web images using incremental learning and hashing approach

B. Steffi Graph *1, M. Ramesh *2

[1] Scholar, Department of CSE, Kalaignar Karunanidhi Institute of Technology, Coimbatore, India.
[2] Assistant Professor, Department of CSE, Kalaignar Karunanidhi Institute of Technology, Coimbatore, India.
Steffime2015@gmail.com[1]

Abstract

Objective: The main objective of this method is to retrieve the image contents from the web server accurately as per the user requirement. This is done by introducing the web page re-ranking mechanism which aims to retrieve the contents content with the knowledge of semantic meaning along with textual information

Method: A novel image re-ranking framework that applies incremental SVM learning algorithm and locality sensitive hashing algorithm is proposed in this work to overcome the problem resides in the existing methodology and retrieve the accurate contents as per the user requirement. This approach is based on the knowledge of universal visual semantic space for highly diverse images.

Results: The performance evaluation of the proposed methodology is done by comparing it with the existing approaches to prove the improvement of proposed approach. The comparison is done by using the performance metrics called the precision, accuracy and recall from which it is proved that the proposed method provides better result than the existing approach.

Conclusion: The finding of this work concludes that the novel re-ranking mechanism retrieves the accurate results than the existing approach with higher accuracy, precision and recall rate.

Keywords: *Search retrieval, keyword query, High diverse images*

1. Introduction

The internet is a place where there are vast resources of information and images. It is a big challenge to find the user intended image from the web. The main reason is user search the image only using a keyword, but it is quite ambiguous to search from pool of image in the web. The user have to look through a lot of irrelevant pages which has more unwanted images and thus it take more time to find the specific images. The most search engines perform keywords based matching and various ranking algorithm. Using visual information to re-rank improve the image search results. This can handle the ambiguity in image search, for example, Apple is a query keyword, and the retrieved images belong to different main categories, such as “Apple Fruit”, “Apple Mac Book”, “Apple tree”, “Green Apple” etc. Within each reference classes there can be many sub classes, and some images are labeled as noise. So adding visual information to image search is important[5,6].

However, interaction should be simple, minimum one- click. One of the image search approach is after query by keyword, user click the image to indicate the query keyword, by incorporating these text and visual information images are re-ranked according to the similarities. Earlier image re-ranking framework has offline and online part. In the offline part keyword along with the image is stored in the index file and also visual features of various features are stored. In the online part user gives the query keyword and selects the image as query image. The given query image is matched with the information stored in the offline part and finally results the re-ranked images.

Earlier work proposes an image re-ranking framework which learns different semantic spaces and an SVM classifier is used for re-ranking the images. It does not cover large images, and also improved performance is not obtained. In order to obtain better accuracy and computation speed a metadata and log data for obtaining co-occurrence information of keywords in user queries. Log data records the events which reduces computation time. In order to update the reference classes over time in an efficient way, incremental SVM learning algorithm is proposed. And finally the matching efficiency is improved by using locality sensitive algorithm. Experimental result provides better result when compared with the existing work.

2. Related works

Jingyu et.al presented online image search re-ranking algorithm which is based on query image and no online training has been done. This work presents Adaptive Similarity which is encouraged by the scheme that a user constantly has a detailed intention while submitting a query image. For instance, when the user submits an image with a full-size face in the center, almost certainly user requires images with similar face. Initially the query image is characterized into one of numerous predefined categories. Within each category, a particular weight schema is found to be combined with the features adaptive to this type of images. When using this image to query, the user intension is reflected by measuring the association between query image and its appropriate similarity measurement and these categories are named as Intentions. The particular weighting schema within each intention category is associated by minimizing the rank loss for every query images on a training set by the present method which is actually modified from Rank Boost technique [1,9].

Xiaoou et.al presented novel Internet image search approach based on visual and textual content based re-ranking. This technique requires only one-click user feedback. Intention specific weight schema is used to unite visual features and to calculate visual adaptive similarity to query images. Despite of human feedback, visual and textual expansions of keywords are incorporated to attain user intention. Expanded keywords are utilized to broaden positive instance images and also widen the image pool to hold additional relevant images. This structure makes it promising for commercial range image search by both visual and text term. The presented image re ranking structure comprises of several steps, which can be enhanced independently or replaced by other methods which is considered consistently effective [2,7].

Yushi Jing et.al [8]presented Visual Rank algorithm, a straight forward method to include the advances made in using network and link investigation for Web document search into image search. Visual Rank appears to diverge from a critical source of information which makes Page Rank more successful: the huge amount of manually produced links on a diverse set of pages. On the other hand, a major quantity of the human coded information is reconvened by two systems. Firstly Visual Rank query dependent is made in which the initial set of images are selected from retrieved answers and human knowledge by means of connecting relevant images to Web pages which is openly initiated into the system. Secondly the image similarity graph is developed based on the general features among images. Those images that detain the common subjects from other images are generally results in higher relevancy [3].

Xiaogang Wang et.al presented semantic space for discovering reference classes for the user keyword. Some reference classes have semantic meanings and their training sets are visually similar. For efficient re-ranking redundant reference classes are removed [4]. The following comparative table shows the merits and demerits of each method in re-ranking category.

Comparison of all these work are depicted in table 1.

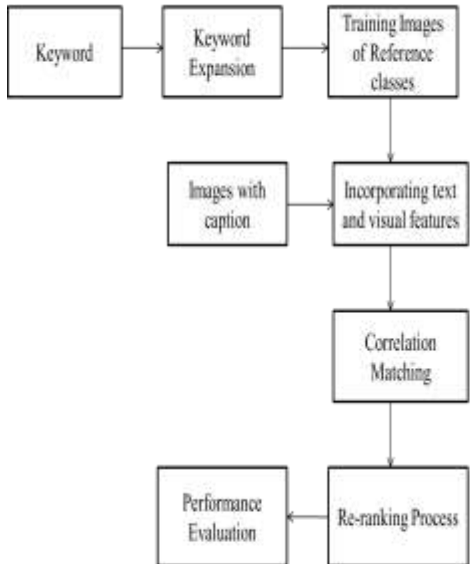
Table 1. Description of related techniques

Authors	Techniques	Merits	Demerits
Jingyu Cui,Fang Wen,Xiaoou Tang	online image search re-ranking algorithm	Overall retrieval performance is improved.	Worst result is obtained for some keywords
Xiaoou Tang, KeLiu, Jingyu Cui, Fang Wen and Xiao gang Wang	Visual and Textual Content Based Re-Ranking	Effective and efficient in Internet image search are designed. Improved ranking is done	Results duplicate images which show similar images to the query. Quality of the re-ranked images are not improved
Yushi Jing, and Shumeet Baluja	Visual Rank	improve the relevancy and diversity of image search results	Ranking and labeling the unlabeled images

3. Re-Ranking of web images

The present work proposes a new image re-ranking framework which incorporates the semantic concepts of both text and visual features which is depicted in figure 1.

Figure 1. Proposed Architecture Diagram



3.1. Keyword expansions

The keywords provided by users tend to be short[10]. They cannot describe the content of images accurately. The query keywords' meanings may be richer than users' expectations. For example, the meanings of the word "apple" include apple fruit, apple computer, and apple iPod. The user may not have enough knowledge on the textual description of target images. For example, if users do not know "gloomy bear" as the name of a cartoon character and they have to input "bear" as query to search images of "gloomy bear". In many cases it is hard for users to describe the visual content of target images using keywords accurately. Google image search provides the "Related Searches" feature to suggest likely keyword expansions. However, even with the same query keywords, the intention of users can be highly diverse and cannot be accurately captured by these expansions "gloomy bear" is not among the keyword expansions suggested by Google "related searches."In our approach, query keywords are expanded to capture users' search intention, inferred from the visual content of query images, which are not considered in traditional keyword expansion approaches. A word w is suggested as an expansion of the query if a cluster of images are visually similar to the query image and all contain the same word w. The expanded keywords better capture users' search intention since the consistency of both visual content and textual description is ensured.

3.2. Reference class selection

For a keyword q, reference classes are defined by finding a set of keyword expansions E (q) most relevant to q. To achieve this, a set of images S(q) are retrieved by the search engine using q as query based on textual information. Keyword expansions are found from words extracted from images in S (q) according to a very large dictionary used by the search engine. A keyword expansion e ∈ E(q) is expected to frequently appear in S(q).For each image I ∈ S(q), all the images in S(q) are reranked according to their visual similarities to I. The T most frequent words $W_I = \{w_I^1, w_I^2, \ldots\ldots w_I^T\}$ among top D re-ranked images (most visually similar to I) are found. $\{w_I^1, w_I^2, \ldots\ldots w_I^T\}$ are sorted by the frequency of words appearing among the D images from large to small. If a word w is among the top ranked image, it has a ranking score $r_I(w)$ according to its ranking order; otherwise $r_I(w) = 0$,

$$r_I(w) = \begin{cases} T - j & w = w_I^j \\ 0 & w \notin W_I \end{cases} \qquad (1)$$

The overall score of a word w is its accumulated ranking scores over all the images

$$r(w) = \sum_{I \in S} r_I(w) \qquad (2)$$

A large $r_I(w)$ indicates that w appears in a good number of images visual similar to I. If w only exists in a small number of images or the images containing w are visually dissimilar to one another, $r_I(w)$ would be zero for most I.

3.3. Metadata generation and log data

The main purpose of metadata is to facilitate in the discovery of relevant information, more often classified as resource discovery. Meta data (the "data about the data") that represent the content of the images in an informative and discriminative fashion. The metadata may include visual signal representation (visual features) of the images acquired using the image analysis techniques combing image processing and computer vision, but also manually inserted and automatically inferred textual annotations. Image search re-ranking stands for the category of techniques that are devised to reorder (refine) the image search results list returned by the text search engine. The refinement aims at a new results list that has better overall relevance to the query than the original one. Since, typically, the information extracted from the visual content of the initially returned images is deployed to derive the re-ranking criteria; image search re-ranking is also often referred to as visual (image search).The log data contains detailed description (time, data and other history) of particular image.

3.4. Incremental SVM learning

For training a SVM "incrementally", on new data by removing all previous (old) data except their support vectors. Consider that user has to be adding an example with existing optimal solution. i.e. reject all the previous data except support vectors. If new point or example is added then initialize the value as 0. And also consider some weight value. . If this is not an optimal solution then update the optimal solution using threshold value. Update the membership of elements at particular time. Finally it will construct the optimal solution.The main building block of the incremental SVM is a procedure for adding one example to an existing optimal solution. When a new point x_c is added, its weight α_c is initially set to 0. If this assignment is not an optimal solution, i.e. when x_c should become a support vector, the weights of other points and the threshold μ must be updated in order to obtain an optimal solution for the enlarged data set. The procedure can be reversed for a removal of an example: its weight is forced to zero while updating weights of the remaining examples and the threshold μ so that the solution obtained with α_c= 0 is optimal for the reduced data set. The saddle point of the problem is given by the Kuhn-Tucker condition

$$g_i: 1 + K_i,: \alpha + \mu y_i \begin{cases} \geq 0, & if \alpha_i = 0 \\ = 0, & if\ 0 < \alpha_i < C \\ \leq 0, & if \alpha_i = C \end{cases} \quad (3)$$

$$\frac{\partial W}{\partial \mu} := y^T \alpha = 0(4)$$

Before an addition of a new example x_c, the Kuhn-Tucker conditions are satisfied for all previous examples. The goal of the weight update in the incremental SVM algorithm is to find a weight assignment such that the Kuhn-Tucker conditions are satisfied for the enlarged data set.

Let us introduce some further notation. Let the set S denote unbounded support vectors ($0 < \alpha_i < C$), the set E denote bounded support vectors ($\alpha_i = C$), and the set O denote non-support vectors ($\alpha_i = 0$); let $R = E \cup O$. These index sets induce respective partitions on the kernel matrix K and the label vector y (we shall use the lower-case letters s, e, o and r for such partitions). By writing out the Kuhn-Tucker conditions (3)–(4) before and after an update$\Delta\alpha$, this work obtains and must satisfy after an update. One can see that $\Delta\alpha_c$is in equilibrium with $\Delta\alpha_s$and μ: any change to $\Delta\alpha_c$must be absorbed by the appropriate changes in $\Delta\alpha_s$and μ in order for the condition to hold. The main equilibrium condition can be further refined as follows. It follows from (1) that Δg_s= 0.

ALGORITHM 1: Incremental Learning

1. Initialize α_c to zero.
2. If $g_c > 0$, terminate (c is not a margin or error vector).
3. If $g_c \leq 0$,apply the largest possible increment α_c so that one of the following conditions occurs:
   (a) $g_c = 0$: Add c to margin set S, update R accordingly, and terminate;
   (b) $\alpha_c = C$: Add c to error set E, and terminate;
   (c) Elements of D^l migrate across S, E, and R : Update membership of elements and , if S changes ,update R accordingly.

and repeat as necessary

3.5. Searching with locality sensitive hashing

Locality sensitive hashing (LSH) is one of the trendiest algorithms for performing approximate search in high dimensions and also able to conclude near neighbors by hashing the query point and retrieving elements stored in memory containing that point. Consider a scenario where giving an image from the database. Then have to mine important feature from this data. By applying Locality Sensitive Hashing - hash tables can be improving the speed of search. Find out the similarity of images from large database images. Then give highly similarity of images also use fast nearest neighbor search for providing most similar images. Finally it provides exact answer for user query fast and efficient. The LSH algorithm relies on the existence of locality-sensitive hash functions. Let$\mathcal{H}$be a family of hash functions mapping $\mathbb{R}^d$to some universe U. For any two points p and q, consider a process in which we choose a function h from $\mathcal{H}$uniformly at random, and analyze the probability that h(p) = h(q). The family $\mathcal{H}$is called locality sensitive (with proper parameters) if it satisfies the following condition. A family $\mathcal{H}$is called (R, cR, P_1, P_2)-sensitive if for any two points $p, q \epsilon \mathbb{R}^d$

$$if \|p-q\| \leq R then Pr_{\mathcal{H}}[h(q)=h(p)] \geq P_1 \quad (5)$$

$$if \|p-q\| \geq cR then Pr_{\mathcal{H}}[h(q)=h(p)] \leq P_2 \quad (6)$$

In order for a locality-sensitive hash (LSH) family to be useful, it has to satisfy P1 >P2 to illustrate the concept, consider the following example. Assume that the data points are binary, that is, each coordinate is either 0 or 1. In addition, assume that the distance between point's p and q is computed according to the Hamming distance. In this case, we can use a particularly simple family of functions $\mathcal{H}$which contains all projections of the input point on one of the coordinates, that is, $\mathcal{H}$contains all functions h_i from $\{0, 1\}^d$ to {0, 1} such that $h_i(p) = p_i$. Choosing one hash function h uniformly at random from $\mathcal{H}$means that h(p) returns a random coordinate of p (note, however, that different applications of h return the same coordinate of the argument). To see that the family His locality-sensitive with nontrivial parameters, observe that the probability $Pr_{\mathcal{H}}[h(q) = h(p)$ is equal to the fraction of coordinates on which p and q agree. Therefore, P_1= 1 – R/d, while P_2= 1 – c R/d. As long as the approximation factor c is greater than 1, we have P1 >P2.

ALGORITHM 2: Locality Sensitive Hashing

Preprocessing:

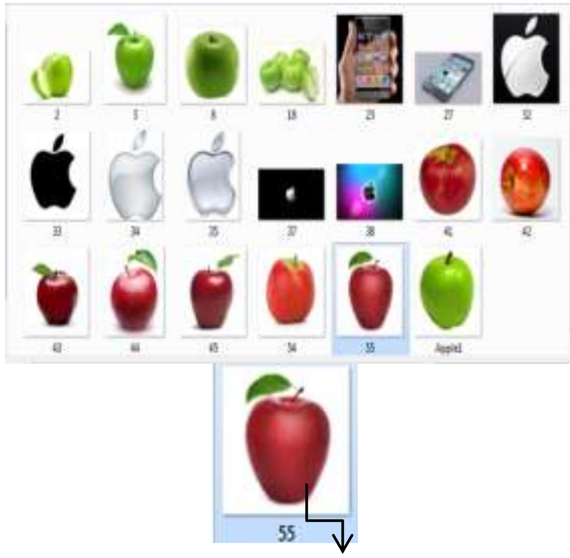
1. Choose L functions g_j,j =1,..L, by setting
   $g_j = (h_{1,j},h_{2,j},\ldots h_{k,j})$, where $h_{1,j},\ldots h_{k,j}$ are chosen at random from the LSH family H.
2. Construct L hash tables , where , for each j = 1,….,L, jth hash table contains the dataset points hashed using the function gj.

Query algorithm for a query point q:

1. For each j = 1,2,..L.
   - i) Retrieve the points from the bucket $g_j(q)$ in the j^{th} hash table.
   - ii) For each of the retrieved point, compute the distance from q to it, and report the point if it is a correct answer (cR – near neighbor for Strategy 1, and R-near neighbor for Strategy 2).
   - iii) (Optional) Stop as soon as the number of reported points is more than L'.

4. Experimental results and discussion

Query Keyword: Apple



Query Image

This section empirically evaluates the proposed system with the existing system. Performance metrics such as Accuracy, precision and recall is measured for image re-ranking with keyword expansion and image re-ranking with query based log keyword expansion.

4.1. Accuracy

The Accuracy of the retrieval rate is measured with the values of the True Negative (TN), True Positive (TP), False Positive (FP), False negative (FN) of the actual class and predicted class results it is defined as follows

$$\text{ACCURACY} = \frac{\text{TP} + \text{TN}}{\text{P} + \text{TN} + \text{FP} + \text{FN}} \qquad (7)$$

4.2. Precision

Precision value is calculated is based on the retrieval of information at true positive prediction, false positive. The data precision is calculated the percentage of positive results returned that are relevant.

$$\text{PRECISION} = \frac{\text{TP}}{(\text{TP} + \text{FP})} \qquad (8)$$

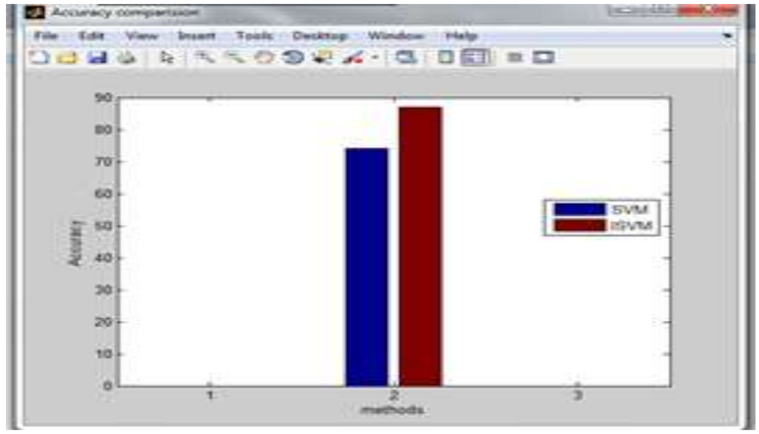
4.3. Recall

Recall value is calculated is based on the retrieval of information at true positive prediction, false negative. Recall is calculated with the percentage of positive results returned that are Recall in this context is also referred to as the True Positive Rate. Recall is the fraction of relevant instances that are retrieved,

$$\text{RECALL} = \frac{\text{TP}}{(\text{TP} + \text{FN})} \qquad (9)$$

The comparison graph for the proposed and existing is shown in figure 5.1

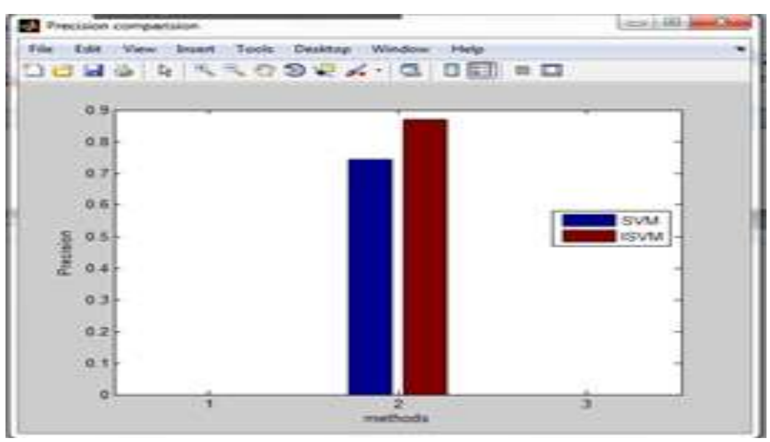
Figure 5.1. Accuracy Comparison



The above graph shows that the accuracy comparison of the methods namely image re-ranking with SVM and image re-ranking with ISVM. The accuracy is measured in % at Y-axis as algorithm and considered the datasets in the X axis. The Accuracy of the re-ranking rate is measured with the values of the True Negative, True Positive, False Positive, False negative. True positive defines a positive test result that accurately reflects the tested-for and activity is analyzed. True negative measures the incorrect data in training and testing, true negative rate is accomplished. False positive result that indicates for a given condition is present when it is not. False negative results indicate that the result appears negative when it should not. From this result re-ranking accuracy is measured with the values of the True Negative, True Positive, False Positive, and False negative with the actual and predicted classes. As a result, the accuracy value of the proposed image re-ranking with I SVM is higher than image re-ranking with SVM.

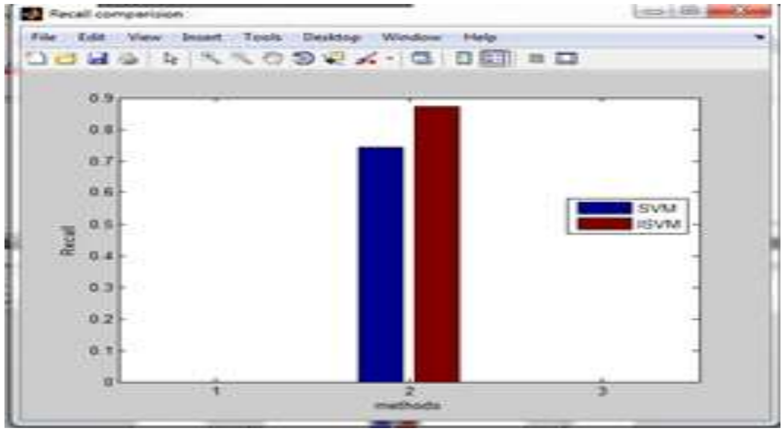
The graph in Figure 5.2 shows that the Precision comparison of the methods namely image re-ranking with SVM and image re-ranking with ISVM. The Precision can be measured at Y-axis as algorithm and considered datasets in the X-axis. Precision value is calculated is based on the retrieval of images at true positive prediction, false positive. In the dataset the value is calculated for these data provides positive result and those result has been considered as relevant. As a result, the Precision value of the image re-ranking with ISVM is higher than image re-ranking with SVM.

Figure 5.2. Precision Comparison



The graph in Figure 5.3 shows that the Recall comparison of the methods namely image re-ranking with SVM and image re-ranking with ISVM. The Recall can be measured at Y-axis as algorithm and considered datasets in the X-axis. Recall value is calculated is based on the retrieval of information at true positive prediction, false negative. In the dataset recall is calculated the percentage of positive results returned that are Recall in this context is also referred to as the True Positive Rate. Recall is the fraction of relevant instances that are retrieved. As a result, the Recall value of the image re-ranking with ISVM is higher than image re-ranking with SVM.

Figure 5.3. Recall Comparison



5. Conclusion

In proposed system, image re-ranking framework is improved by extracting the keyword features of metadata and log data. In addition to visual and textual features, Keyword expansion defines the reference classes and incorporates other metadata and log data. From this features, the co-occurrence information of keywords in user queries can be obtained in log data. In order to update the reference classes over time in an efficient way, incremental SVM learning algorithm is proposed. And finally the matching efficiency is improved by using Locality sensitive hashing algorithm which efficiently matches the smallest semantic signatures and possible to make them more compact and enhances the matching efficiency. But the major limitation of the exact incremental learning is its memory requirement, since the set of support vectors must be retained in memory during the entire learning. Future work will include further investigation of properties of incremental SVM such as numerical stability and their utility for tracking the values of generalization bounds.

6. References

1. Xiao gang Wang, Member, Shi Qiu, Ke Liu, Xiao Tang. Web image Re-ranking using query-specific semantic signatures. *IEEE*. 2014; 36 (4), 810-823.
2. J. Cui, F. Wen, X. Tang. Real time google and live image search Re-ranking. Proceedings of 16th ACM International Conference, Multimedia. 2008.
3. X. Tang, K. Liu, J. Cui, F. Wen, X. Wang. Intent search: Capturing user intention for one-click internet image search. *IEEE*. 2012; 34(7), 1342-1353.
4. Y. Jing, S. Baluja. Visual rank: Applying page rank to large-scale image search. *IEEE*. 2008; 30(11), 1877-1890.
5. G. Cauwenberghs, T. Poggio. Incremental and Decremental Support Vector Machine Learning. Proceedings of Advances in Neural Information Processing Systems (NIPS). 2001.
6. A.Andoni, P. Indyk. Near-optimal hashing algorithms for approximate nearest neighbor in high dimensions. *Communication of the ACM*. 2008; 51(1), 117-122.
7. X. Wang, K. Liu, X. Tang. Query-specific visual semantic spaces for web image Re-ranking. Proceeding of IEEE Conference Computer Vision and Pattern Recognition (CVPR), 2011; 857-864
8. J. Lu, J. Zhou, J. Wang, X. Hua, S. Li. Image search results refinement via outlier detection using deep contexts. Proceedings of IEEE Conference Computer Vision and Pattern Recognition (CVPR), 2012.
9. Y. Kuo, W. Cheng, H. Lin, W. Hsu. Unsupervised semantic feature discovery for image object retrieval and tag refinement. *IEEE*. 2012; 14 (4), 1079-1090.
10. L. Yang, A. Hanjalic. Supervised reranking for web image search. Proceedings of ACM International Conference Multimedia, New York. 2010; 183-192.

The Publication fee is defrayed by Indian Society for Education and Environment (iSee). www.iseeadyar.org

***Citation**:*

B. Steffi Graph, M. Ramesh. An efficient Re-ranking of web images using incremental learning and hashing approach. *Indian Journal of Innovations and Developments*. 2014; 3 (4), 85-92