

# META-HEURISTICS BASED ARF OPTIMIZATION FOR IMAGE RETRIEVAL

Darsana B<sup>1</sup> and Dr. Rajan Chattamvelli<sup>2</sup>

<sup>1</sup>The Oxford College of Engineering, Bangalore, India.

darsana.b@live.com

<sup>2</sup>Periyar Maniammai University, Thanjavur, India.

cvrajan@gmail.com

## ABSTRACT

*The proposed approach avoids the semantic gap in image retrieval by combining automatic relevance feedback and a modified stochastic algorithm. A visual feature database is constructed from the image database, using combined feature vector. Very few fast-computable features are included in this step. The user selects the query image, and based on that, the system ranks the whole dataset. The nearest images are retrieved and the first automatic relevance feedback is generated. The combined similarity of textual and visual feature space using Latent Semantic Indexing is evaluated and the images are labelled as relevant or irrelevant. The feedback drives a feature re-weighting process and is routed to the particle swarm optimizer. Instead of classical swarm update approach, the swarm is split, for each swarm to perform the search in parallel, thereby increasing the performance of the system. It provides a powerful optimization tool and an effective space exploration mechanism. The proposed approach aims to achieve the following goals without any human interaction - to cluster relevant images using meta-heuristics and to dynamically modify the feature space by feeding automatic relevance feedback.*

## KEYWORDS

*Content based Image Retrieval, Latent Semantic Indexing, Meta-heuristics, Semantic gap.*

## 1. INTRODUCTION

Image retrieval plays a vital role in Image Archives, Domain Specific Collections, Enterprise Collections, Personal Collections, and the Web. An image retrieval system is a software system for browsing, searching and retrieving images from a large database of digital images. Retrieval process includes, submitting a query, and extracting images that best matches the user request. The query may be text-based or image-based. Image retrieval is broadly classified into two divisions - Image meta search and Content Based Image Retrieval (CBIR).

Image meta search deals with search of images based on associated metadata such as keywords, text, etc. Inherent problem is that annotating thousands of images is a questionable process, and the search results are highly dependent on the subjectivity of human perception. Based on the human perception, the search tags may vary, which brings up evident differences in the retrieved results.

The paramount challenge of semantic gap is bridged by Content-based Image Retrieval [1], which is one of the hot spots in the field of image retrieval in recent years. CBIR filters out images

based on similarities in their contents to a user-supplied query image. In other words, it extracts images' visual information by analysing the inherent data in images. However, the algorithm of a single visual image feature represents only a partial property of the image [2], so search results have not been very good. In the extraction of combined features, traditional methods mostly extract manifold features from the original image. This has the disadvantage that the feature vector is too big and computational complexity is too high, especially for large databases.

The performance of these image-centric retrieval systems is not satisfactory primarily due to the mismatch between the user's implied concept and the low-level visual features. In order to narrow this gap, relevance feedback was introduced as an interactive tool in CBIR [3].

Although relevance feedback can significantly improve the retrieval performance in CBIR systems, its applicability still suffers from two inherent problems. First, user interaction for providing feedback is time consuming and tiring. Secondly, previous feedback results are typically not retained in the system.

This paper is organised as follows - In Section 2 we briefly summarise the related work on image retrieval, relevance feedback and meta-heuristics. The proposed approach is described in Section 3. In Section 4, we present experimental setup, results and discussions. Finally, Section 5 draws the conclusions and identifies future research directions.

## **2. RELATED WORK**

### **2.1. Content Based Image Retrieval**

In recent years, CBIR systems have gained much popularity. Huge number of images makes it difficult to locate the image searched for, especially when the search is based on the esthetic value of the image. Content Based Image Retrieval (CBIR) uses a query image to retrieve all matching images within a specified threshold. Major area of focus in this, is the semantic gap, i.e., understanding the subjective meaning of the visual query. The less the semantic gap is, the results are more accurate. Most of the research has concentrated on feature extraction of an image, e.g., QBIC [4]- which queries images based on their color, texture and shape, VisualSeek [5] - A Fully Automated Content-Based Image Query System, SIMPLicity [6] - Semantics- Sensitive Integrated Matching for Picture Libraries, and Blobworld [7] - Image segmentation using expectation-maximization algorithm and applies the same to image querying.

### **2.2. Relevance Feedback**

In order to narrow down the inconsistency of textual annotation and semantic gap of visual query, the proposed approach uses relevance feedback in image retrieval. The basic Relevance Feedback mechanism relies on iteratively asking the user to discriminate between relevant and irrelevant images on a given set of results. A binary Relevance Feedback is used to train neural network systems as in PicSOM [8] and in Bordogna and Pasi [9].

Applicability of Relevance feedback suffers from two vital problems - User interaction for providing feedback is time consuming and it is a tiring process. In order to curb these cons, automatic relevance feedback is used. This eliminates the user interaction totally in the feedback process, thus saving time and energy to a great extent.

### 2.3. Meta-Heuristics

In the last years, the development of optimization algorithms have been inspired and influenced by natural and biological behaviours. Bio-inspired meta-heuristic optimization approaches provided new ways to achieve nearly-optimal solutions in highly nonlinear, multidimensional solution spaces, with lower complexity and faster convergence than traditional algorithms.

A bio-inspired stochastic optimization algorithm called Particle Swarm Optimization (PSO) was introduced in the field of computational intelligence by Kennedy and Eberhart [10] in 1995. PSO is a population based meta-heuristic optimization technique with stochastic nature, inspired by social behaviour of bird flocking or fish schooling.

The system is initialised with a population of random solutions and searches for optima by updating generations. The potential solutions, called particles, fly through the problem space by following the current optimum particles. PSO has been successfully applied as an efficient optimization tool in image classification [11].

### 3. PROPOSED APPROACH

This paper proposes an avant-garde approach of Meta-Heuristics Based ARF Optimization for Image Retrieval, which avoids the semantic gap in image retrieval by combining automatic relevance feedback and a stochastic algorithm. A visual feature database is constructed from the image database, using combined feature vector calculated from colour moments, taking the mean, standard deviation and skewness, and Gabor filter into account.

A combined feature vector is generated from colour moments (9 dimensions) & Gabor filters for the query image. Visual feature database construction is done offline, and feature vector for query image is constructed online. This way, the response time is not affected by the overhead of feature vector computation for database images. Very few fast-computable features are included in this step, because this is applied sequentially to all images present in the database.

Query image is mapped into a feature vector and the distance between the query and image is calculated. The system ranks the whole dataset according to a minimum distance criterion. The distance is the sum of weighted Euclidean distances between pairs of feature vectors – feature vector of the database image and feature vector of the query image. The Weighted Euclidean Distance WED(X,Y) is calculated as follows:

$$WED(X,Y) = 1/S \cdot \sum_{s=1}^S (X_s - Y_s)^2 \cdot W_s^k \quad \text{---- (1)}$$

where, X,Y are feature vectors corresponding to the query image and the images in the database, S is the dimension of the feature vector,  $W_s^k$  is the weight associated with feature vector, k is the iteration number. If  $k=1$ ,  $W_s^k = 1$  for all values of s.

For the first iteration, the weight of every feature is identical. From the second iteration, the weight of each feature changes according to feature reweighing factor. This will be reflected in the distance between the query image and the database images.

Based on the computed distance, the nearest images are retrieved from the database and are routed to automatic relevance feedback. Here the images are split into relevant and irrelevant subsets.

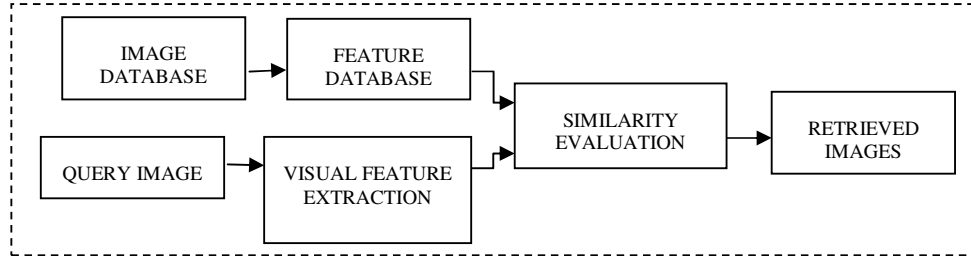


Figure 1. Image Retrieval

For the retrieved images, first automatic relevance feedback based on Latent Semantic Indexing (LSI) is generated. Here, LSI is applied in both textual and visual (image key) feature space. The textual feature space is constructed by using the keywords related to the image.

If there is no associated keyword with an image, then textual feature space calculation is automatically ignored by setting the value of  $\alpha$  to 0.

$$TSim(q,i) = (q.i) / (|q| |i|) \quad \text{---- (2)}$$

where,  $q$  is term associated with the query image and  $i$  is the terms associated with the images in the database.

The visual feature space comprises of a feature vector, which is a combination of Color histogram bins (32 dimensions), Edge histogram bins (16 dimensions), wavelet texture energy values (18 dimensions).

$$VSim(q,i) = (q.i) / (|q| |i|) \quad \text{---- (3)}$$

where,  $q$  is feature vector of the query image and  $i$  is the feature vector of the image in the database.

The combined similarity of textual and visual feature spaces is evaluated and the images are labelled as relevant or irrelevant, based on the similarity value.

$$VTSim(q,i) = \alpha . Tsim + (1 - \alpha) . Vsim \quad \text{---- (4)}$$

where,  $\alpha$  is a constant value. To use only the textual feature space of the image,  $\alpha$  value is set to 1, and to use only the visual feature space of the image, when no keywords are associated with the image,  $\alpha$  value is set to 0. Here  $Tsim$  represents the textual similarity measure, and  $Vsim$  represents visual similarity measure. Accordingly, relevant and irrelevant image subsets are created, which will be progressively populated across iterations, based on the change in weights of individual features, thus changing the distance between the query image and the database images.

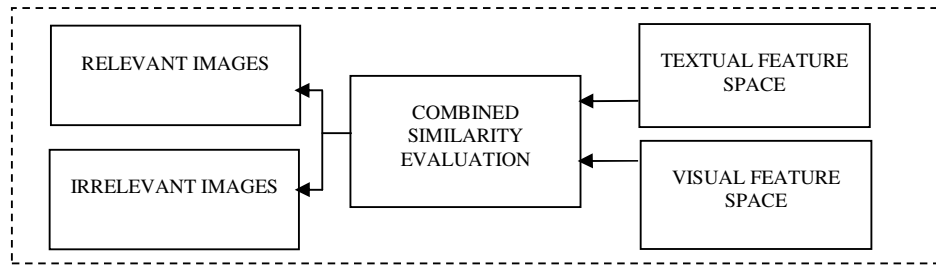


Figure 2. Automatic Relevance Feedback

### 3.1. Particle Swarm Optimizer

The feedback drives a feature re-weighting process [5] and is routed to the particle swarm optimizer. The feature reweighting process determines the importance of individual features in the retrieval process adaptively. Based on the reweighted feature value, the importance of each feature is set in the forth-coming iterations.

Particle swarm optimizer is used to grasp image's semantics through optimized iterative learning. It provides a powerful optimization tool and an effective space exploration mechanism [6]. A very preliminary version of PSO-CBIR was presented in [7]. After the relevant images are computed through automatic relevance feedback, the swarm is initialized as follows: number of swarm particles  $p$ , is initialized corresponding to the number of retrieved images, and each particle is associated to one of the  $P$  nearest neighbours of the original query.

Personal best, i.e.,  $pbest$  is the feature vector of the retrieved images associated to the particle. Global best, i.e.,  $gbest$  is the feature vector of query image. Each particle has its own  $pbest$  value, and the  $gbest$  value is shared among all the particles in the swarm.

PSO's fitness function represents the effectiveness of the solution reached by the swarm particles. Fitness value of each particle is calculated and the swarm is ranked according to the lowest fitness value first criterion. The image associated with the particle with least fitness value is more relevant to the query image given by the user.

From the second iteration, the swarm evolves as follows: The position of the swarm particles is updated in each iteration, starting from the second iteration. The position of the particle  $p_n^k$  is given by,

$$p_n^k = p_n^{k-1} + v_n^k \quad \text{----- (5)}$$

where,  $p_n^{k-1}$  is the position of the particle in previous iteration and  $v_n^k$  is the random speed vector for the current iteration. The speed of the random vector keeps minimizing, as it reaches convergence. This ensures deep exploration of more related images. The images corresponding to the updated position of the particles are retrieved.

The  $gbest$  is updated as an image of the 'relevant' set. For every relevant image, the sum of distances from other relevant images is calculated; then, the image resulting nearest to others is chosen as  $gbest$ . The particles in the swarm, evolves in each iterations, by moving towards the  $gbest$ .

After that, a new ranking is calculated based on the weighted Euclidean distance between the query image and the images corresponding to the new position of the particles in the swarm. Based on the updated swarm ranking, the nearest images are proposed to collect a new feedback from automatic relevance feedback. The swarm is split as a multiple of two with respect to current number of swarms. Each swarm now starts searching in parallel, which increases the overall performance of the system. The updated results are taken to proceed with the next iteration till convergence.

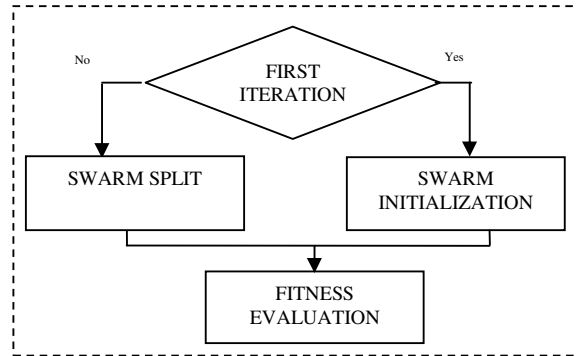


Figure 3. Particle Swarm Optimization

#### 4. EXPERIMENTAL SETUP AND RESULTS

The experimental data comprises of collection of generic images from the Corel image database (<http://www.corel.com>) [13], SIMPLicity[6], [14] and the web images. We prepared five classes composed of 20 categories – (car, ship, cartoon, trucks), (birds, animals, sunflower, insects), (desert, city night, mountains, farm lands), (kites, butterflies, rivers, sceneries) and (base ball, basket ball, balls, play ground). The number is limited as compared to the huge number of images used in large-scale applications, but it provides clear classification of images, so as to evaluate the performance of the proposed method.

A combined feature vector is computed offline for the database images. This feature vector semantically represents the images in the database. The query image is chosen by the user, and the retrieval processes for the relevant images are triggered. Based on the Euclidean distance between the combined feature vector, the relevant images are retrieved by the system and the first iteration commences.

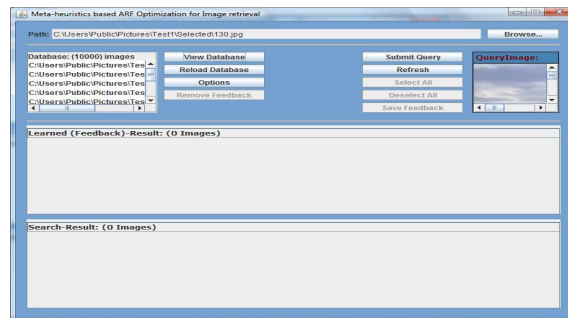


Figure4. Query Image Selection

The retrieved images are passed for automatic relevance feedback using Latent Semantic Indexing. Here textual feature space and visual feature space are computed for the retrieved images. The combined feature space is used to provide relevant and irrelevant subsets of images, with respect to the query image.

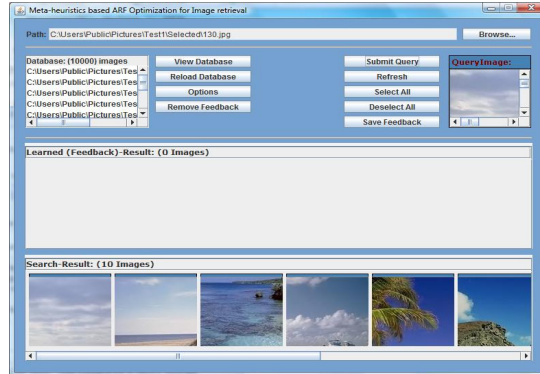


Figure5. Set of retrieved images

In the Particle Swarm Optimizer, the particles are positioned corresponding to the first set of retrieved images and swarm initialization is done. The fitness value of each particle is calculated. The particle with least fitness value represents the best fit image for the query, and that is maintained as global best for that iteration.

From the second iteration, the swarm is split into multiple of two, with respect to the current number of swarms, with each swarm having its own global best value. From then on, the particles start moving towards their corresponding swarm's global best.

The position of the particles is updated in each iteration, based on a random vector value. The random vector value [10] is calculated based on an inertial weight factor which is fixed in the range [0.2,0.7] and two positive constants  $C_1$ ,  $C_2$  - called acceleration coefficients, aimed at pulling the particle towards the position related to the personal best and global best ( $C_1=C_2=2$ ). The final automatic feedback results can be saved optionally, to make use of it in the future.

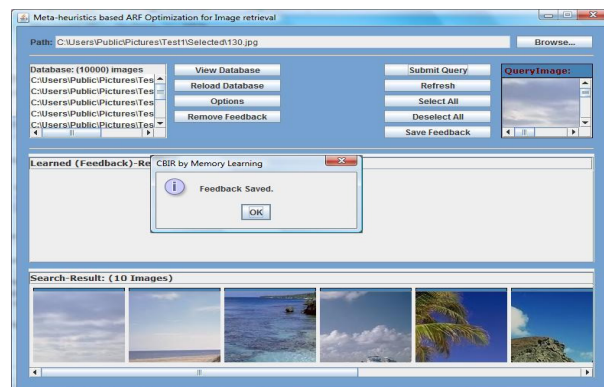


Figure6. Final Feedback saving for Retrieved Relevant Images

The important difference of the proposed approach from typical optimization problems is that the data are not completely available at the beginning, but they are collected from the automatic feedback across iterations in an incremental manner.

As iterations progress, a user need input lesser information. This unsupervised version makes the convergence faster, as compared to standard implementations, since the learning procedure is driven automatically without any user intervention.

The performance of the system is projected in terms of precision and recall. Precision is calculated as the number of retrieved relevant images over the feedback images and recall is computed as percentage of relevant images retrieved across all iterations with respect to the number of class samples.

The maximum average precision turns to be 0.80 and recall is 0.76. For all given classes of images, the proposed approach gives similar and satisfactory results, which gives a clear picture of the effectiveness of the approach for different classes of images. The following graph gives the average precision - recall computation with the given classes.

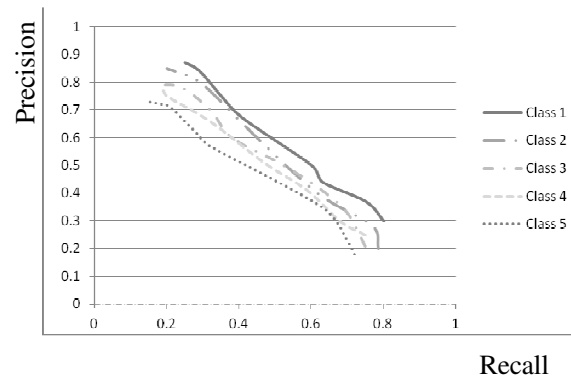


Figure 7. Average Precision Vs Recall

## 5. CONCLUSION

The initial Image Retrieval uses fast and easily computable features, to reduce computational cost. The features selected for Automatic Relevance Feedback are capable of retrieving in-depth low level features of the image. Feature reweighing emphasizes the most discriminating parameters. The image retrieval problem is formulated as an optimization problem, and is solved using particle swarm optimization.

It takes into account the characteristics of relevant and irrelevant images, as points of attraction and repulsion, thus performing an effective retrieval. The proposed approach achieves the following goals without any human interaction – clustering relevant images using meta-heuristics and dynamically modifies the feature space by feeding automatic relevance feedback.

## REFERENCES

- [1] Song Yan, “Research of Image Retrieval Based on color and texture feature”, Computer application and soft computing, 2007, 9(2),pp.42-50.
- [2] Ye Yu-guang, “Research of image retrieval based on fusing with multi-character”, Hua Qiao university, 2007, pp.14-36.



- [3] Y. Rui, T.S. Huang, and S. Mehrotra, "Relevance feedback a powerful tool in interactive content-based image retrieval", IEEE Transactions on Circuits and Systems for Video Technology, vol. 8, no. 5, pp. 644-655, 1998.
- [4] W. Niblack, R. Barber, W. Equitz, M. Flickner, E. Glasman, D. Pektovic, P. Yanker, C. Faloutsos, and G Taubin, "The QBIC project: Querying images by content using color, texture and shape," Proceedings of SPIE Storage and Retrieval for Image and Video Databases, San Jose, CA, 1994.
- [5] J. R. Smith and S.-F. Chang, "VisualSEEK: A Fully Automated Content-Based Image Query System," Proceedings of the Fourth ACM International Conference on Multimedia '96, Boston, MA, 1996.
- [6] J. Z. Wang, J. Li, and G. Wiederhold, "SIMPLIcity: Semantics- Sensitive Integrated Matching for Picture Libraries," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 23, pp. 947-963, 2001.
- [7] C. Carson, S. Belongie, H. Greenspan, and J. Malik, "Blobworld: Image segmentation using expectation-maximization and its application to image querying," IEEE Transaction on Pattern Analysis and Machine Intelligence.
- [8] M. Koskela, J. Laaksonen, and E. Oja, "Use of image subsets in image retrieval with self-organizing maps," in Proc. Int. Conf. Image and Video Retrieval (CIVR), 2004, pp. 508-516.
- [9] G. Bordogna and G. Pasi, "A user-adaptive neural network supporting a rule-based relevance feedback," Fuzzy Sets Syst., vol. 82, no. 9, pp. 201-211, 1996.
- [10] J. Kennedy and R. C. Eberhart, "Particle swarm optimization," in Proc. IEEE Conf. Neural Networks IV, Piscataway, NJ, 1995.
- [11] K. Chandramouli, "Particle swarm optimization and self-organizing maps based image classifier," in Proc. 2nd Int. Workshop Semantic Media Adaptation and Personalization, 2007, pp. 225-228.
- [12] Tse-Wei Chen, Yi-Ling Chen, and Shao-Yi Chien , "Fast Image Segmentation and Texture Feature Extraction for Image Retrieval", 2009 IEEE 12th International Conference on Computer Vision Workshops, ICCV Workshops.
- [13] S. C. Hoi, M. R. Lyu, and R. Jin, "A unified log-based relevance feedback scheme for image retrieval," IEEE Transaction on Knowledge and Data Engineering, vol. 18, no. 4, pp. 509-524, Apr. 2006.
- [14] Jia Li, James Z. Wang, "Automatic linguistic indexing of pictures by a statistical modeling approach," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 25, no. 9, pp. 1075-1088, 2003.

## Authors

Darsana B is working as a Senior Lecturer in The Oxford College of Engineering, Bangalore. She has around 4 years of experience in this profession. She has completed her B. Tech in Information Technology, M.E in Computer Science and Engineering and currently pursuing her Ph D in the area of Image Retrieval under the guidance of Dr. Rajan Chattamvelli. So far she has presented 4 papers in different conferences and guiding different academic projects related to Image Processing.



Rajan Chattamvelli has more than 15 years experience in software development, teaching and research. He has taught computer science at various colleges and universities in India and Cyprus. His areas of research include data mining, algorithms design, information retrieval, object oriented technologies, parallel computing, design patterns and statistics. He is the author of 3 books and more than 16 research papers. He has also presented several papers at national and international conferences. He can be contacted at cvrajan@gmail.com or through skype at rajan.chattamvelli

