Aggregate Similarity Queries in Relevance Feedback Methods for Content-based Image Retrieval*

Humberto L. Razente, Maria Camila N. Barioni, Agma J. M. Traina, Caetano Traina Jr.
Computer Sciences Department – ICMC/USP
Caixa Postal 668 – 13560-970 – São Carlos – SP – Brazil
{ hlr, mcamila, agma, caetano }@icmc.usp.br

ABSTRACT

Content-based image retrieval techniques rely on automatic features extracted from images to process similarity queries. Usually low-level features are extracted, and when they are used to compare images stored in a database to a reference image (through single center selection queries), they often lack the ability to convey to the users what they understand as similarity. To deal with the gap between what the user expects and what the system can automatically provide, relevance feedback techniques have been employed. In this paper we present a generalization of the single center similarity queries over data in metric spaces, taking into account both range and k-nearest neighbors. Allowing a query to include multiple query centers, it straightforwardly attends the relevance feedback requirements. Thus, we analyze how well our new approach contribute to relevance feedback methods for content-based image retrieval.

Categories and Subject Descriptors

H.2.4 [**Database Management**]: Systems—*Multimedia databases*; H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval—*Relevance feedback*

Keywords

Aggregate Similarity Queries, Relevance Feedback, Content-based Image Retrieval

1. INTRODUCTION

Similarity queries are very useful to search complex data. A similarity query searches the database looking for elements similar to a reference element (query center) according to a certain similarity measure, ranking the results based on the distances to the query center. The similarity comparisons usually require the data domains to be represented in metric spaces, which embody both spatial data with fixed dimensions as well as non-dimensional data [17]. A metric space is defined by a pair $\langle \mathbb{S}, \delta() \rangle$, where $\mathbb{S}$ is the data domain and $\delta()$ is a dissimilarity function $\delta : \mathbb{S} \times \mathbb{S} \rightarrow R^+$ that complies with the symmetry, non-negativity, identity, and triangle inequality properties. Some studies also consider non-metric similarity, such as in [1]. However, when a dataset is represented in a metric space, it is possible to create index structures (metric access methods) that allow the optimization of similarity queries.

Complex data (such as images, audio or video) require extracting features that are used in place of the data element when performing the comparisons. The features are usually the result of mathematical algorithms, resulting in low level features. Considering the image domain, the features are usually based on color, texture and shape, as in Content-Based Image Retrieval (CBIR). However, there exists a semantic gap between the low level features and the human interpretation subjectivity. To deal with the semantic gap, relevance feedback techniques have been developed. In these techniques, positive and/or negative examples are informed by the user to allow the system to derive a more precise representation of the user intent [18]. The new representation of the user intent can be achieved through query point movement or multiple point movement techniques. Furthermore, implicit feedback techniques have also been developed for web search engines. In these techniques, the system learns from search results provided by the user and takes advantage of this information to adapt ranking functions. One way to tell to the system what is the user's intention is specifying, in the same query, other elements besides the query center, which are positive or negative examples of the intend answer. This representation is based on multiple query centers.

Previous works have employed aggregate functions to rank the elements from a dataset based on multiple query centers. Considering metric spaces, [15] proposed the Falcon technique to rank images based on an aggregate function up to a threshold ξ. The works presented in [14] and [10] propose the use of an aggregate function to rank low-dimensional spatial data respectively by range and by k-nearest neighbors.

In this paper, we propose the generalization of the two most common similarity queries in metric spaces – the range

*This work has been supported by FAPESP (São Paulo State Research Foundation), by CAPES (Brazilian Coordination for Improvement of Higher Level Personnel) and by CNPq (Brazilian National Council for Supporting Research).

SAC'08 March 16-20, 2008, Fortaleza, Ceará, Brazil

and the k-nearest neighbor query – defining the Aggregate Similarity Query, which retrieves elements based on their composite similarity regarding multiple centers. An aggregate similarity query can be limited either by an aggregate threshold ξ – resulting in the Aggregate Range Query – or by a quantity k of elements – resulting in the Aggregate k-Nearest Neighbor Query. We also discuss their usage for positive and negative weighted relevance feedback for CBIR, showing results over real image datasets.

The remainder of the paper is structured as follows. Section 2 summarizes existing related works. Section 3 presents the fundamental concepts regarding the proposed aggregate similarity query. A representative set of experiments is presented in Section 4. Finally, Section 5 gives the conclusions of this paper.

2. RELATED WORK

Relevance feedback is a real-time learning strategy that adapts the answer of a retrieval system exploring the user interaction. It is a process that takes advantage of the information fed back by the user about the relevance of the answers returned by a given query to automatically adjust the answer to the next queries [3]. Thus, it aims at obtaining a better approximation of the user's expectations and preferences.

The typical steps of a relevance feedback cycle in image retrieval systems are: (1) the system returns the initial answers of a query-by-example; (2) the user judges the answers returned informing a degree of relevance to a set of images; (3) the system learns the user intention based on the user feedback. When the user poses a new query, the system changes its search rules to adapt the answer to the learned intention. The cycle is repeated until the user is satisfied with the results, as shown in Figure 1 [9].

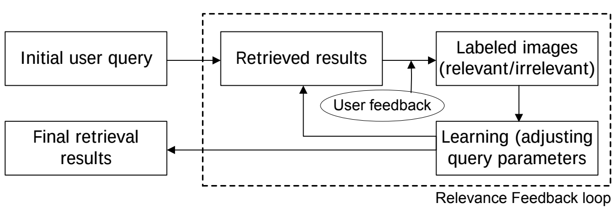

Figure 1: CBIR with relevance feedback.

Considering step 2, several algorithms with different approaches have been proposed. There are some algorithms that assume the feedback either as positive or negative [16, 4], others that only deal with positive feedback [2] and others that accept varying degrees of relevance for positive and negative examples [19]. There is also a number of techniques that can be employed in step 3. According to the strategy employed, these techniques can be divided into two main categories: query point movement, and multiple point movement.

The query point movement techniques consider that a query is represented by a single query center. Therefore, at each user interaction cycle, the strategy estimates an ideal query center in the query space, moving the center towards the relevant examples and away from the irrelevant ones, as in Figure 2. When dealing with multidimensional data, these techniques usually perform a refinement of the dissimilarity function using the information obtained from the user to decide which dimensions should be emphasized and which should have their influence diminished. The refinement is done assigning weights to all dimensions of the feature vector [13, 7].

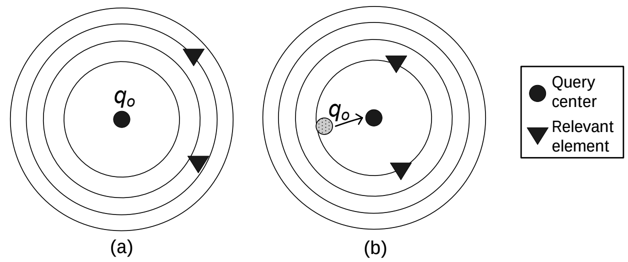

Figure 2: Query point movement example. (a) Initial query. (b) New query point.

Considering that the feedback may be composed of positive and negative examples, the Rocchio's technique [12] is one of the most common techniques employed to compute the query point movement. However, the Rocchio's technique can be employed when the feature vectors have a fixed dimension. It is based on the formula expressed by Equation 1, where Q is the query vector and Q' is the resultant vector, D'_R and D'_N are respectively the positive and negative examples, $N_{R'}$ and $N_{N'}$ are the number of positive and negative examples in D'_R and D'_N and α, β and γ are selected constants, obtained experimentally. The Rocchio's relevance feedback technique maximizes the difference among the query center Q and the images selected as non-relevant D'_N and minimizes the difference among the query center Q and the images selected as relevant D'_R. In this way, all relevant and non-relevant elements are employed to move the query center Q, adding the normalized differences of D'_R and subtracting the normalized differences of D'_N, resulting in a new element Q'. It is important to note that the query point movement creates a new feature vector that may not correspond to an image from the dataset.

$$Q' = \alpha Q + \beta \left(\frac{1}{N_{R'}} \sum_{i \in D'_R} D_i \right) - \gamma \left(\frac{1}{N_{N'}} \sum_{i \in D'_N} D_i \right) \quad (1)$$

On the other hand, multiple point movement techniques utilize multiple query centers to represent a query, as shown in Figure 3. The strategy of these techniques consists in assigning the elements informed as relevant in clusters, where each cluster is represented by the closest element to the cluster center. The new centers are called representatives, and are used to execute a multiple center query that calculates a single contour to cover all the representative elements. In this type of query, it is common to use the number of elements associated to each center as its weight. Thus, the distance of one query center to the representatives in a multiple center query corresponds to a weighed combination of the individual distances among the associated elements and the representatives. This strategy was employed in [6] and [11]. In addition to these techniques, there are also strategies based on multiple point movement, where the elements labeled as relevant form disjunctive clusters, thus allowing the execution of disjunctive queries [15, 8].

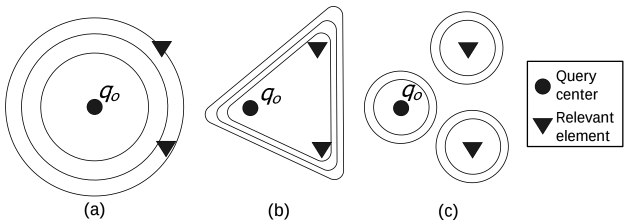

Figure 3: Multiple point movement example. (a) Initial query. (b) Query expansion based on weights. (c) Multiple query center technique.

3. AGGREGATE SIMILARITY QUERIES

In order to search by similarity, the dataset elements can be represented as vectors in a multidimensional space (e.g. color features extracted from images), by a non-fixed number of values (e.g. fingerprints, where the number of deltas and endings may differ from one person to another) or by values in an adimensional space (e.g. DNA sequences represented as texts), depending on the type of application. It is worth to note that multidimensional data compared with the Minkowski distance functions are special cases of metric spaces. As in this paper we aim at generalizing the concept of aggregate similarity queries, we focus on the metric space model because it is less restrictive and embodies all of these cases. Provided with adequate dissimilarity functions, the metric space also includes the multidimensional space.

Aggregate similarity queries are based on aggregate similarity predicates, and we define an aggregate similarity predicate as follows. Given a set of elements $S \subset \mathbb{S}$, where $\mathbb{S}$ is the domain of elements from a metric space $\langle \mathbb{S}, \delta() \rangle$, a set of query centers $Q \subset \mathbb{S}$ and a similarity aggregation function $d_g()$ that calculates the aggregate similarity of each element $s_i \in S$ regarding its similarity measured by $\delta()$ to every element $s_q \in Q$, an aggregate similarity predicate $\wp(\langle d_g(), Q, \ell \rangle) : S$ retrieves every element $s_i \in S$ whose similarity aggregated by the aggregation function $d_g()$ does not exceed the limit ℓ.

There are two basic types of similarity predicates: those limiting the answer based on a given similarity threshold ξ and those limiting the answer based on the number k of elements in the answer. If a proper aggregation function $d_g()$ is employed, the predicates of the well-known similarity range and k-nearest neighbor queries turn out as special cases of the $\wp$ aggregate similarity predicates, where the set of query centers has only one element $Q = \{s_q\}$ and the limit ℓ is either the range radius or the number of neighbors, respectively.

Relevance feedback techniques are usually similarity queries expressed by the similarity selection operation $\hat{\sigma}$ based on similarity predicates. The similarity selections exhibit properties distinct from those of the traditional selections (for example, they are not commutative), so we use the $(\hat{\sigma})$ symbol instead of the traditional σ. Therefore, we define the following two kinds of similarity queries.

Definition 1: Aggregate Range Query (ARq): given a maximum aggregate query distance ξ, a similarity aggregation function $d_g()$ and a set of query centers Q, the query ARq retrieves every element $s_i \in S$, such that $d_g \leq \xi$. An aggregate range selection can be expressed in relational algebra as $\hat{\sigma}_{\left(ARq\langle d_g(), Q, \xi \rangle\right)} S$.

Definition 2: Aggregate k-Nearest Neighbor Query (kANNq): given an integer value $k \geq 1$, the query $kANNq$ retrieves the k elements that result in the minimum value of the similarity aggregation function $d_g()$ from the query centers Q in S. An aggregate k-nearest neighbor selection can be expressed as $\hat{\sigma}_{\left(kANNq\langle d_g(), Q, k \rangle\right)} S$.

Following the definition of metric spaces, the dissimilarity function $\delta()$ can be any function comparing pairs of elements $s_i, s_j \in \mathbb{S}$ that follows the symmetry, non-negativity, identity, and triangle inequality properties. However, as the set of query centers Q may have more than one element, the distance $\delta(s_i, s_q)$ from each query center $s_q \in Q$ to the element $s_i \in S$ must be evaluated to calculate the similarity aggregation function $d_g : P(\mathbb{S}) \times \mathbb{S} \rightarrow R^+$ between s_i and the set of query centers $Q \subset P(\mathbb{S})$. The similarity predicate uses the resulting value to rank the elements in S with respect to Q.

There may be several alternatives to define the similarity aggregation function $d_g()$. We consider that it is generated by Equation 2, where $\delta()$ is a dissimilarity function, Q is the set of query centers, $s_i \in S$ is an element of the dataset S, w_q is the weight corresponding to element s_q and $g \in \mathbb{R}^+$ is a non-zero real value we call the grip factor of the similarity aggregation function. Considering Equation 2, the Aggregate Range Queries and the Aggregate k-Nearest Neighbor Queries applied over single query centers correspond to the traditional Range Queries and the k-Nearest Neighbor Queries respectively.

$$d_g(Q, s_i) = \sqrt[g]{\sum_{s_q \in Q} [\ \delta(s_q, s_i)^g \cdot w_q\ ]} \tag{2}$$

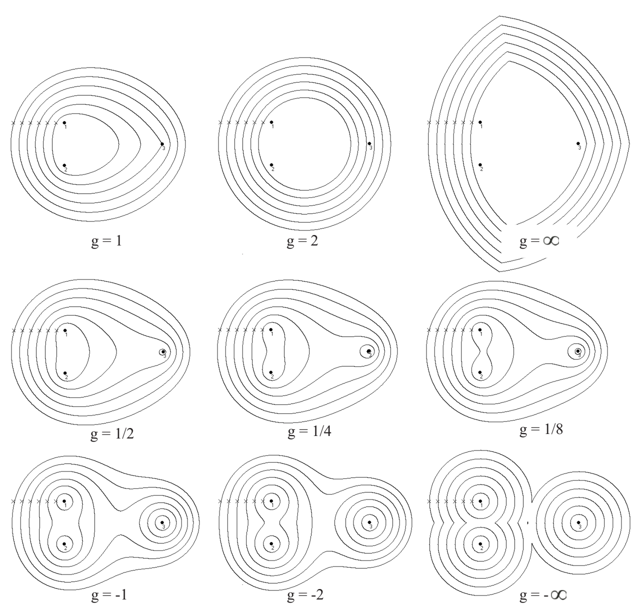

Figure 4: The effect of the grip factor g in an Euclidean 2-dimensional space, considering $Q = \{q_1, q_2, q_3\}$.

Figure 4 presents the effect of the grip factor g in a 2-dimensional Euclidean space, considering Q composed of the 3 shown query centers. Each line represents the geometric place where Equation 2 has the same value, thus they are isolines in the 2D Euclidean space, or isohypersurfaces in a generic metric space. Each isoline represents a different covering radius, allowing the definition of both range and k-limited aggregate queries. Notice that for $g < 1$ they may generate disjunctive regions.

Weights can be used to model the perception of users in relevance feedback methods. The methods may let the users to set the weights or automatically compute them based on the user's feedback. Figure 5 shows the covering regions in a 2-dimensional space for a set of three query centers using the grip factor $g = 1$. In this figure, isoline (a) represents the covering region considering all weights equal to 1 (positive feedback) and isoline (b) represents the covering region setting the weight $w_2 = -0.5$ (negative feedback) for the query center q_2 and $w_1 = w_3 = 1$ to the query centers q_1 and q_3.

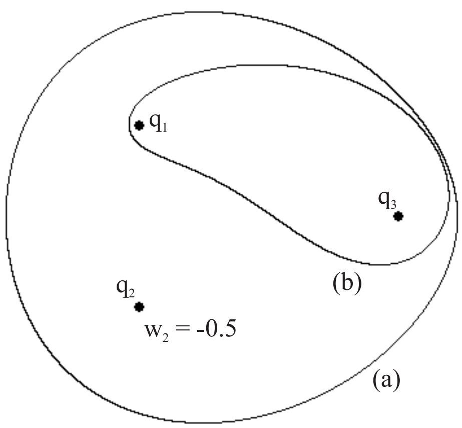

Figure 5: Aggregate range ($g = 1$) covering regions in a 2-dimensional Euclidean space for the set of query centers $Q = \{q_1, q_2, q_3\}$. Isoline (a) represents the covering region considering all weights equal to 1. Isoline (b) represents the covering region considering the weight $w_2 = -0.5$ for the query center q_2.

4. EXPERIMENTS

In order to show the effectiveness of our approach, we present two representative sets of experiments chosen from those we have performed with several datasets. The image dataset employed in the experiments is the Amsterdam Library of Object Images (ALOI) [5], a color image collection of one-thousand small objects, recorded for scientific purposes in several configurations. The ALOI Illumination Color dataset is composed of 12 different illumination photos of each object, resulting in 12,000 images. For the sake of the simplicity of the aggregate similarity queries analysis, the feature extractor employed returns the values of the red, green and blue in a 256-scale histogram for each image (256 dimensions) and the distance function employed is the Manhattan ($L_1()$), although specific applications, such as content-based medical image retrieval, should employ more sophisticated feature extractors and distance functions. Figure 6 shows a sample of 10 elements of this dataset.

Figure 6: A sample of the images of the ALOI Illumination Color dataset.

In the experiments, we evaluate the effect of the grip factor g regarding the precision *versus* recall plots of 300-nearest neighbor queries and the first three relevance feedback cycles. Notice that, in the precision *versus* recall graphs, the closer a curve is to the top, the better is the method under evaluation. We present herein the results based on nearest neighbors due to the fact that the k-limited queries are the ones that most relevance feedback methods employ. All feedback cycles were run for aggregate 300-nearest neighbor queries. To evaluate the correctness of each image retrieved, we used the object portrayed in the image as a class attribute, therefore performing a supervised automated evaluation of the algorithm. This is a common configuration employed to perform a large number of tests over the algorithm under evaluation. For each query, the images fed back and the images considered for the computation of precision are those related to the same class in the dataset. The plots show the average result for 100 randomly chosen query images.

4.1 Positive Feedback

In the first set of experiments, we evaluate positive feedback. It explored the grip factors $g = 2$, $g = 1$, $g = 1/2$ and $g = 1/4$. Figure 7 shows the precision *versus* recall results. The feedback increased the precision in the three first cycles, and after them, comparing the results for up to 76% of recall, they resulted in the precisions of 78.4% for $g = 2$, 85.2% for $g = 1$, 90.5% for $g = 1/2$ and 93.4% for $g = 1/4$. High precision values up to high recall levels are the desirable results.

The same experiment was executed with the Rocchio's technique, setting the positive feedback constant for two values of β, $\beta = 0.5$ and $\beta = 1.0$, executing the initial k-nearest neighbor queries and performing query point movements in each cycle. These settings enable direct comparison of the results, as the experiments were run under the same conditions. Figure 8 shows the precision *versus* recall graphics. The feedback also increased the precision in the three first cycles, and after that, comparing the results for up to 76% of recall, they resulted in the precisions of 67.8% for $\beta = 0.5$ and of 77.8% for $\beta = 1.0$. Other values of β were also tested, but did not result in better precision than for $\beta = 1.0$. As it can be noticed, even the first cycle of our proposed algorithm achieves a better precision than the best one obtained using the Rocchio's technique (86.8% for $g = 1/4$ *versus* 74.3% for $\beta = 1.0$, both at 76% of recall).

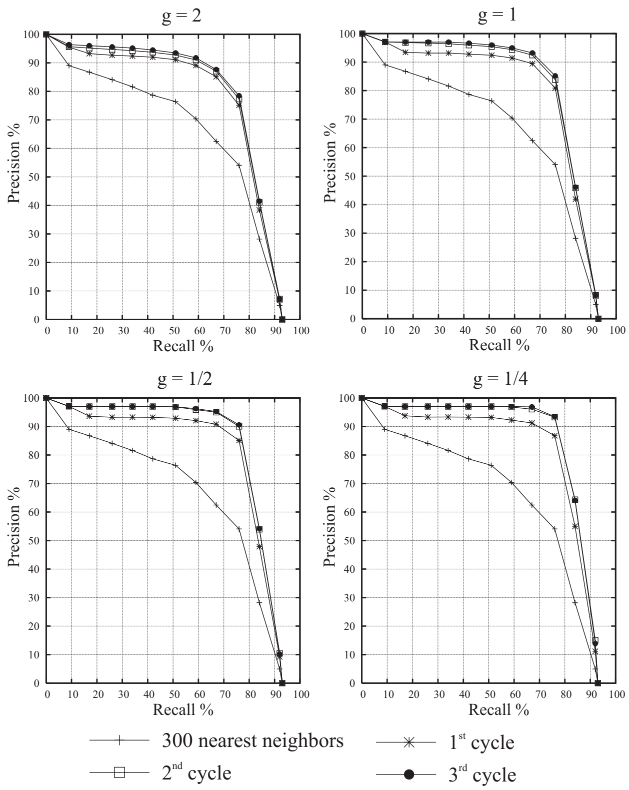

Figure 7: ALOI Illumination Color dataset precision *versus* recall graphics of k-nearest neighbor queries and the first 3 cycles. Positive feedback. (a) $g = 2$ (b) $g = 1$ (c) $g = 1/2$ (d) $g = 1/4$.

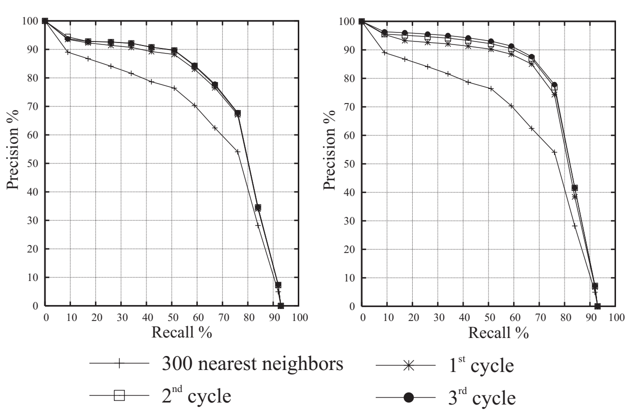

Figure 8: ALOI Illumination Color dataset precision *versus* recall graphics of k-nearest neighbor queries and the first 3 cycles. Positive feedback. (a) Rocchio $\beta = 0.5$ (b) Rocchio $\beta = 1.0$.

4.2 Positive and Negative Feedback

In the second set of experiments, we evaluate the positive and negative feedbacks. The negative feedback images were chosen from the resultant set of k images of each query among those with class distinct from the query center, in a number not larger than 33% of the images selected as positive feedback (images of the same class), by setting the weight $w = -0.5$. It also explored the grip factors $g = 2$, $g = 1$, $g = 1/2$ and $g = 1/4$. Figure 9 shows the precision *versus* recall results. The feedback increased the precision in the first three cycles, and after them, comparing the results for up to 76% of recall, they resulted in the precisions of 70.9% for $g = 2$, 76.9% for $g = 1$, 81.7% for $g = 1/2$ and 85.8% for $g = 1/4$.

Again, we run the same experiment with the Rocchio's formula, combining the positive feedback constant $\beta = 0.5$ and $\beta = 1.0$ with the negative feedback constant $\gamma = 0.5$, executing the initial k-nearest neighbor queries and performing query point movements in each cycle. Figure 10 shows the precision *versus* recall graphics. The feedback also increased the precision in the first three cycles, and after them, comparing the results for up to 76% of recall, they resulted in the precisions of 65.3% for $\beta = 0.5$ and $\gamma = 0.5$ and of 73.8% for $\beta = 1.0$ and $\gamma = 0.5$. Other combinations of values of β and γ were also tested, but did not result in better precision than for $\beta = 1.0$ and $\gamma = 0.5$. As occurred in the positive feedback experiment, even the first cycle of our proposed algorithm achieves a better precision than the best one obtained using the Rocchio's technique (81.5% for $g = 1/4$ *versus* 76.2% for $\beta = 1.0$, both at 76% of recall).

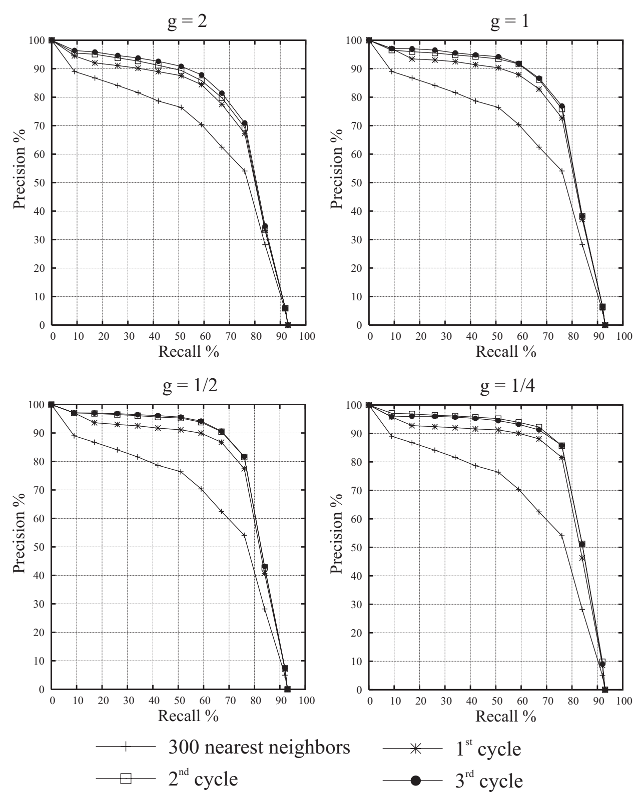

Figure 9: ALOI Illumination Color dataset precision *versus* recall graphics of k-nearest neighbor queries and the first 3 cycles. Positive and negative feedback. (a) $g = 2$ (b) $g = 1$ (c) $g = 1/2$ (d) $g = 1/4$.

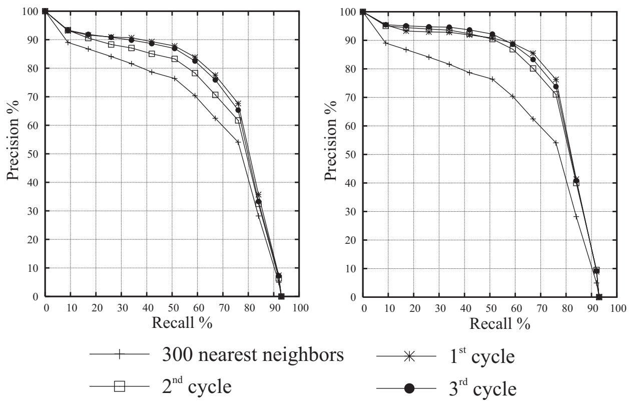

Figure 10: ALOI Illumination Color dataset precision *versus* recall graphics of k-nearest neighbor queries and the first 3 cycles. Positive and negative feedback. (a) Rocchio $\beta = 0.5$ $\gamma = 0.5$ (b) Rocchio $\beta = 1.0$ $\gamma = 0.5$.

5. CONCLUSION

This paper presented the aggregate similarity queries in metric spaces, which can be seen as the generalization of the most common queries (range and k-nearest neighbors) for multiple query centers, and their usefulness for relevance feedback. As shown in the experiments section, the aggregate similarity queries are useful in real applications and present better behavior than the Rocchio's technique, one of the best and most used techniques for relevance feedback in CBIR. Further work includes the development of optimization algorithms to perform these queries using metric access methods.

6. REFERENCES

[1] G.-H. Cha. Non-metric similarity ranking for image retrieval. In *Int'l Conf. on Database and Expert Systems Applications (DEXA), Lecture Notes in Computer Science (LNCS) Volume 4080*, pages 853–862, Krakow, Poland, 2006. Springer.

[2] Y. Chen, X. Zhou, and T. S. Huang. One-class SVM for learning in image retrieval. In *IEEE Int'l Conf. on Image Processing (ICIP)*, pages 34–37, Thessaloniki, Greece, 2001.

[3] N. Doulamis and A. Doulamis. Evaluation of relevance feedback schemes in content-based in retrieval systems. *Signal Processing: Image Communication*, 21(4):334–357, 2006.

[4] M. Ferecatu, M. Crucianu, and N. Boujemaa. Improving performance of interactive categorization of images using relevance feedback. In *IEEE Int'l Conf. on Image Processing (ICIP)*, pages 1197–1200, Genova, Italy, 2005.

[5] J.-M. Geusebroek, G. J. Burghouts, and A. W. M. Smeulders. The Amsterdam library of object images. *Int'l Journal of Computer Vision*, 61(1):103–112, 2005.

[6] K. A. Hua, N. Yu, and D. Liu. Query decomposition: A multiple neighborhood approach to relevance feedback processing in content-based image retrieval. In *IEEE Int'l Conf. on Data Engineering (ICDE)*, pages 84–94, Atlanta, GA, 2006.

[7] Y. Ishikawa, R. Subramanya, and C. Faloutsos. Mindreader: Query databases through multiple examples. In *Int'l Conf. on Very Large Databases (VLDB)*, pages 218–227, New York, 1998.

[8] D.-H. Kim and C.-W. Chung. Qcluster: relevance feedback using adaptive clustering for content-based image retrieval. In *ACM Int'l Conf. on Management of Data (SIGMOD)*, pages 599–610, San Diego, CA, 2003.

[9] Y. Liu, D. Zhang, G. Lu, and W.-Y. Ma. A survey of content-based image retrieval with high-level semantics. *Pattern Recognition*, 40(1):262–282, 2007.

[10] D. Papadias, Y. Tao, K. Mouratidis, and C. K. Hui. Aggregate nearest neighbor queries in spatial databases. *ACM Transactions on Database Systems (TODS)*, 30(2):529–576, 2005.

[11] K. Porkaew, S. Mehrotra, and M. Ortega. Query reformulation for content based multimedia retrieval in MARS. In *IEEE Int'l Conf. on Multimedia Computing and Systems (ICMCS)*, pages 747–751, Florence, Italy, 1999.

[12] J. J. Rocchio. *Relevance Feedback in Information Retrieval*. The SMART Retrieval System: Experiments in Automatic Document Processing. Prentice-Hall, Englewood Cliffs, New Jersey, 1971.

[13] Y. Rui, T. S. Huang, M. Ortega, and S. Mehrotra. Relevance feedback: A power tool for interactive content-based image retrieval. *IEEE Transactions on Circuits and Video Technology*, 8(5):644–655, 1998.

[14] Y. Tao and D. Papadias. Range aggregate processing in spatial databases. *IEEE Transactions on Knowledge and Data Engineering*, 16(12):1555–1570, 2004.

[15] L. Wu, C. Faloutsos, K. Sycara, and T. R. Payne. Falcon: Feedback adaptive loop for content-based retrieval. In *Int'l Conf. on Very Large Databases (VLDB)*, pages 297–306, Cairo, Egypt, 2000.

[16] P.-Y. Yin, B. Bhanu, K.-C. Chang, and A. Dong. Integrating relevance feedback techniques for image retrieval using reinforcement learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(10):1536–1551, 2005.

[17] P. Zezula, G. Amato, V. Dohnal, and M. Batko. *Similarity Search: The Metric Space Approach (Advances in Database Systems)*. Springer-Verlag, 2005.

[18] X. S. Zhou and T. S. Huang. Relevance feedback in image retrieval: A comprehensive review. *Multimedia Systems*, 8(6):536–544, 2003.

[19] Z.-H. Zhou, K.-J. Chen, and H.-B. Dai. Enhancing relevance feedback in image retrieval using unlabeled data. *ACM Transactions on Information Systems (TOIS)*, 24(2):219–244, 2006.