Query-Sensitive Similarity Measure for Content-Based Image Retrieval

Zhi-Hua Zhou Hong-Bin Dai
National Laboratory for Novel Software Technology
Nanjing University, Nanjing 210093, China
{zhouzh, daihb}@lamda.nju.edu.cn

Abstract

Similarity measure is one of the keys of a high-performance content-based image retrieval (CBIR) system. Given a pair of images, existing similarity measures usually produce a static and constant similarity score. However, an image can usually be perceived with different meanings and therefore, the similarity between the same pair of images may change when the concept being queried changes. This paper proposes a query-sensitive similarity measure, `Qsim`*, which takes the concept being queried into account in measuring image similarities, by exploiting the query image as well as the images labeled by user in the relevance feedback process. Experimental comparisons to state-of-the-art techniques show that* `Qsim` *has superior performance.*

1 Introduction

In a typical CBIR setting, a user poses an example image, i.e. *query*, and asks the system to bring out relevant images from the database. A main difficulty here is the gap between high-level semantics and low-level image features, due to the rich content but subjective semantics of an image. Relevance feedback has been shown as a powerful tool for bridging this gap [4, 10]. In relevance feedback, the user has the option of labeling a few images according to whether they are relevant to the target or not. The labeled images are then given to the CBIR system as complementary queries so that more images relevant to the query can be retrieved from the database.

In fact, the retrieval engine of a CBIR system can be regarded as a machine learning process, which attempts to train a learner to classify the images in the database into two classes, i.e. *relevant* or *irrelevant*. In CBIR a basic assumption is that the *relevant* images should be more similar to the query than the *irrelevant* images do. Therefore, similarity measure is one of the keys of a high-performance CBIR system, and many endeavors have been devoted to the design of appropriate image similarity measures [2, 3, 5, 7, 8].

Existing similarity measures usually produce a static and constant similarity score for a given pair of images. However, since images can be perceived with different meanings, the similarity between the same pair of images may change when the query changes, as Figure 1 illustrates.

This paper advocates that similarities between images should be able to be regularized by the concept being queried. In this way, the user preference, or subjectivity, in the ambiguous image meanings could be captured to some extent. Inspired by advances in text-based information retrieval [6], this paper proposes `Qsim`, a *query-sensitive* similarity measure, which takes the concept being queried into account in measuring the similarities between different images, by exploiting the original query image as well as the images labeled by user in the relevance feedback process. Moreover, considering the *asymmetrical training sample* problem in CBIR, i.e., relevant images can be regarded as belonging to the same concept class but irrelevant images are usually belonging to different concept classes, `Qsim` employs a scheme that exploits relevant and irrelevant images differently. Empirical comparisons to state-of-the-art techniques exhibit the superiority of `Qsim`.

2 QSIM

Let $\mathcal{P}$ and $\mathcal{N}$ denote the labeled relevant and irrelevant images, respectively. Initially, $\mathcal{P}$ contains only the query image and $\mathcal{N}$ is an empty set. Later, $\mathcal{P}$ and $\mathcal{N}$ are enriched by the images labeled by user in the relevance feedback process. Let $|\mathcal{P}|$ and $|\mathcal{N}|$ denote the sizes of $\mathcal{P}$ and $\mathcal{N}$, respectively. Since an image is usually represented by an n-dimensional feature vector, for convenience of discussion, here the elements in $\mathcal{P}$ and $\mathcal{N}$ are regarded as feature vectors directly.

Let z^+ denote a query, which is of course relevant to the concept being queried. For two feature vectors $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ in the database, this paper defines the similarity between $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ given z^+ as:

$$Sim(\boldsymbol{x}_i, \boldsymbol{x}_j | \boldsymbol{z}^+) = Sim(\boldsymbol{x}_i, \boldsymbol{x}_j) \times Sim(\boldsymbol{x}_i, \boldsymbol{x}_j, \boldsymbol{z}^+) \quad (1)$$

(a) ***image-1*** **(b)** ***image-2*** **(c)** ***image-3*** **(d)** ***query-1:*** *forests* **(e)** ***query-2:*** *wild cat*

Figure 1. An illustration of the motivation of the paper: When *query-1* **is posed,** *image-2* **should be more similar to** *image-1* **than to** *image-3* **since both** *image-1* **and** *image-2* **are relevant to** *forests* **but** *image-3* **is irrelevant to** *forests*; **however, when** *query-2* **is posed,** *image-2* **should be more similar to** *image-3* **than to** *image-1* **since both** *image-2* **and** *image-3* **are relevant to** *wild cat* **but** *image-1* **is irrelevant to** *wild cat*.

where $Sim\left(\boldsymbol{x}_i, \boldsymbol{x}_j\right)$ and $Sim(\boldsymbol{x}_i, \boldsymbol{x}_j, \boldsymbol{z}^+)$ are defined in Eqs. 2 and 3, respectively. It is evident that Eq. 2 is a similarity measure based on Euclidean distance, where ξ is used for avoiding zero denominator. Here Eq. 2 is called as a *base similarity measure* since Eq. 1 and following similarity measures are defined based on it.

$$Sim\left(\boldsymbol{a}, \boldsymbol{b}\right) = \frac{1}{\sqrt{\sum_{k=1}^{n} \left|\boldsymbol{a}_k - \boldsymbol{b}_k\right|^2 + \xi}} \tag{2}$$

$$Sim(\boldsymbol{a}, \boldsymbol{b}, \boldsymbol{c}) = \frac{Sim(\boldsymbol{a}, \boldsymbol{c}) + Sim(\boldsymbol{b}, \boldsymbol{c})}{2} \tag{3}$$

It is easy to prove that *Eq. 1 can behave well for the scenario shown in Figure 1*:

Proof. Let $\boldsymbol{u}_1$, $\boldsymbol{u}_2$ and $\boldsymbol{u}_3$ denote the feature vectors corresponding to Figures 1(a) to 1(c), respectively, and for simplicity, assume that initially $Sim(\boldsymbol{u}_1, \boldsymbol{u}_2) = Sim(\boldsymbol{u}_2, \boldsymbol{u}_3)$. Now, given query $\boldsymbol{q}$ corresponding to Figure 1(d), $Sim(\boldsymbol{u}_1, \boldsymbol{q}) > Sim(\boldsymbol{u}_3, \boldsymbol{q})$ and $Sim(\boldsymbol{u}_2, \boldsymbol{q}) > Sim(\boldsymbol{u}_3, \boldsymbol{q})$ since both $\boldsymbol{u}_1$ and $\boldsymbol{u}_2$ are relevant to $\boldsymbol{q}$ but $\boldsymbol{u}_3$ is irrelevant to $\boldsymbol{q}$. Then, according to Eq. 3, $Sim(\boldsymbol{u}_1, \boldsymbol{u}_2, \boldsymbol{q}) > Sim(\boldsymbol{u}_2, \boldsymbol{u}_3, \boldsymbol{q})$. According to Eq. 1, $Sim(\boldsymbol{u}_1, \boldsymbol{u}_2|\boldsymbol{q}) > Sim(\boldsymbol{u}_2, \boldsymbol{u}_3|\boldsymbol{q})$. Thus, given $\boldsymbol{q}$, $\boldsymbol{u}_2$ becomes more similar to $\boldsymbol{u}_1$ than to $\boldsymbol{u}_3$, although initially it was equally similar to $\boldsymbol{u}_1$ and $\boldsymbol{u}_3$. □

In fact, Eq. 1 realizes the idea that two images will become more similar if both of them are similar to a relevant image. This is not difficult to understand since a relevant image should have characterized the concept being queried to some extent, and images similar to the relevant image should share some positive characteristics. Thus, when there is a set of relevant images, an *individual relevance score* of image $\boldsymbol{x}$ to the target concept from the view of a relevant image $\boldsymbol{z}_t^+ \subset \mathcal{P} = \{\boldsymbol{z}_k^+ | k = 1, \cdots, |\mathcal{P}|\}$ can be estimated according to:

$$Score_t^+(\boldsymbol{x}) = \max_{i \in \{1, \cdots, |\mathcal{P}|\}} Sim(\boldsymbol{x}, \boldsymbol{z}_i^+ | \boldsymbol{z}_t^+) \tag{4}$$

By aggregating the individual relevance scores obtained from the view of every relevant image in $\mathcal{P}$, the relevance score of $\boldsymbol{x}$ to the concept being queried can be derived:

$$Score^+(\boldsymbol{x}) = \frac{1}{|\mathcal{P}|} \sum_{t=1}^{|\mathcal{P}|} Score_t^+(\boldsymbol{x}) \tag{5}$$

Thus, given the query and some relevant images provided by the user, the relevance scores of all the images in the database to the concept being queried can be estimated according to Eq. 5, which can be used to rank the images to be returned to the user.

However, Eq. 5 only considers relevant images, while the user can label some irrelevant images in the relevance feedback process. Considering that irrelevant images have different properties from relevant images, it is not appropriate to use Eq. 1 to incorporate the influences of irrelevant images on the similarity.

Given an irrelevant image $\boldsymbol{z}^-$, this paper uses Eq. 6 to measure the similarity between images $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$:

$$Sim(\boldsymbol{x}_i, \boldsymbol{x}_j | \boldsymbol{z}^-) = \frac{Sim(\boldsymbol{x}_i, \boldsymbol{x}_j)}{Sim(\boldsymbol{x}_i, \boldsymbol{x}_j, \boldsymbol{z}^-)} \tag{6}$$

where $Sim\left(\boldsymbol{x}_i, \boldsymbol{x}_j\right)$ and $Sim(\boldsymbol{x}_i, \boldsymbol{x}_j, \boldsymbol{z}^-)$ are defined in Eqs. 2 and 3, respectively.

It is easy to prove that *Eq. 6 enables two images become more similar if both of them are dissimilar to an irrelevant image*:

Proof. Let $\boldsymbol{u}_1$, $\boldsymbol{u}_2$ and $\boldsymbol{u}_3$ denote the feature vectors corresponding to three images, respectively, and for simplicity, assume that initially $Sim(\boldsymbol{u}_1, \boldsymbol{u}_2) = Sim(\boldsymbol{u}_2, \boldsymbol{u}_3)$. Now, given an irrelevant image $\boldsymbol{v}$, and assume it is dissimilar to $\boldsymbol{u}_1$ and $\boldsymbol{u}_2$ but similar to $\boldsymbol{u}_3$. That is, $Sim(\boldsymbol{u}_3, \boldsymbol{v}) > Sim(\boldsymbol{u}_1, \boldsymbol{v})$ and $Sim(\boldsymbol{u}_3, \boldsymbol{v}) > Sim(\boldsymbol{u}_2, \boldsymbol{v})$. Then, according to Eq. 3, $Sim(\boldsymbol{u}_1, \boldsymbol{u}_2, \boldsymbol{v}) < Sim(\boldsymbol{u}_2, \boldsymbol{u}_3, \boldsymbol{v})$. According to Eq. 6, $Sim(\boldsymbol{u}_1, \boldsymbol{u}_2|\boldsymbol{v}) > Sim(\boldsymbol{u}_2, \boldsymbol{u}_3|\boldsymbol{v})$. Thus, given $\boldsymbol{v}$, $\boldsymbol{u}_2$ becomes more similar to $\boldsymbol{u}_1$ than to $\boldsymbol{u}_3$, although initially it

was equally similar to $\boldsymbol{u}_1$ and $\boldsymbol{u}_3$. □

When there are a set of irrelevant images as well as a set of relevant images, an individual relevance score of image $\boldsymbol{x}$ to the target concept from the view of an irrelevant image $\boldsymbol{z}_s^- \subset \mathcal{N} = \{\boldsymbol{z}_k^- | k = 1, \cdots, |\mathcal{N}|\}$ can be estimated according to:

$$Score_s^-(\boldsymbol{x}) = \underset{i \in \{1,\cdots,|\mathcal{P}|\}}{average} \; Sim(\boldsymbol{x}, \boldsymbol{z}_i^+ | \boldsymbol{z}_s^-) \qquad (7)$$

It is worth noting that the operator *average* is used in Eq. 7 instead of the operator *max* used in Eq. 4. This is because that the roles played by irrelevant images are different from that played by relevant ones. Indeed, if two database images are very similar to a relevant image, then these two database images have big chances to be relevant, no matter whether these two database images are similar to other relevant images or not. So, Eq. 4 uses *max*. On the other hand, if two database images are very dissimilar to an irrelevant image, it is not sufficient to conclude that these database images have big chances to be relevant, since they might be similar to some other irrelevant images. In other words, the influences of all the irrelevant images should be considered. So, Eq. 7 uses *average*.

By aggregating the individual relevance scores obtained from the view of every irrelevant image in $\mathcal{N}$, the relevance score of $\boldsymbol{x}$ to the concept being queried can be derived:

$$Score^-(\boldsymbol{x}) = \frac{1}{|\mathcal{N}|} \sum_{s=1}^{|\mathcal{N}|} Score_s^-(\boldsymbol{x}) \qquad (8)$$

Finally, by considering the contribution of relevant images as well as irrelevant images, a final relevance score of $\boldsymbol{x}$ to the concept being queried becomes:

$$Score(\boldsymbol{x}) = \frac{1}{\mathrm{Z}_+} Score^+(\boldsymbol{x}) + \frac{1}{\mathrm{Z}_-} Score^-(\boldsymbol{x}) \qquad (9)$$

where Z_+ and Z_- are normalization factors which make the contribution of $Score^+(\boldsymbol{x})$ and $Score^+(\boldsymbol{x})$ equally important. In other words, the values of $Score^+(\boldsymbol{x})$ and $Score^-(\boldsymbol{x})$ are mapped into the same range before they are summed to obtain $Score(\boldsymbol{x})$.

Given a set of relevant images and a set of irrelevant images, which can be obtained by collecting the query image as well as images labeled by user in the relevance feedback process, the `Qsim` method uses Eq. 9 to estimate the relevance score of every database image to the concept being queried, and then returns the images by ranking them descendingly according to their relevance scores.

`Qsim` uses the relevant images as well as irrelevant images to determine the relevance score. Such a scheme has been adopted by many existing techniques, e.g. [2]. Note that the key difference between `Qsim` and them lies in the fact that in these techniques the similarity between the same pair of images is always the same, while in `Qsim` the similarity between the same pair of images can be different for different queries.

3 Experiments

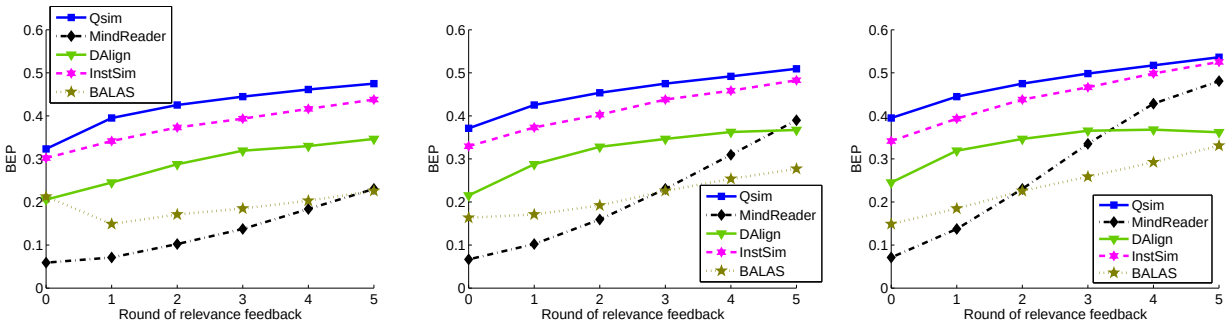
In the experiments, `Qsim` is compared with a representative earlier method, `MindReader` [3], and three recent state-of-the-art techniques, i.e. $\texttt{DA}_{lign}$ [8], `InstSim` [2] and `BALAS` [9]. For techniques which require the input of pairs of similar images and pairs of dissimilar images, pairs of relevant images are used as pairs of similar images, while pairs of a relevant image and an irrelevant image are used as pairs of dissimilar images.

Here every image is represented by a 67-dimensional feature vector, which involves color, texture and shape features. Twenty classes of COREL images each has 100 images are used. Each time a class of images are used as relevant images while the remaining ones are regarded as irrelevant. After obtaining the initial query, five rounds of relevance feedbacks are performed. In each round the user can label F (= 4, 6, or 8) number of images as the feedback, half of which are relevant images while the other half are irrelevant ones. For each query, the relevance feedback process is repeated for five times with different users. Moreover, the whole process is repeated for five times with different queries. The average results are recorded for each class of images. The experiments are conducted on a Pentium 4 machine with 3.00GHz CPU and 1GB memory.

PR-graph is a performance measure popularly used in CBIR, which depicts the relationship between *precision* and *recall* of a specific retrieval system. However, in order to exhibit the changes of the retrieval performance in relevance feedback, a single PR-graph is not enough. Instead, a series of PR-graphs each corresponding to a round of relevance feedback have to be used. Due to the page limit, PR-graphs are not presented in this paper.

BEP-graph was proposed recently by Zhou et al. [11]. By definition, if the precision and recall are tuned to have an equal value, then this value is called the Break-Event-Point (BEP) of the system. The higher the BEP, the better the performance. Through connecting the BEPs after different rounds of relevance feedbacks, BEP-graph is obtained, where the horizontal axis enumerates the round of relevance feedback while the vertical axis tells the BEP value. The geometrical BEP-graphs are presented in Figure 2.

Moreover, a quantitative measure, *effectiveness* (η_S) [1], is employed, where S denotes the number of relevant images the user wants to retrieve. The bigger the value of η_S, the better the retrieval performance. The geometrical effectiveness are tabulated in Table 1 where the best performance at each round of relevance feedback has been boldfaced.



(a) $F = 4$ **(b)** $F = 6$ **(c)** $F = 8$

Figure 2. Geometrical BEP-graphs of `Qsim`, `MindReader`, $\mathtt{DA}_{lign}$, `InstSim` and `BALAS`.

Table 1. Geometrical η_S of `Qsim` (Q), `MindReader` (M), $\mathtt{DA}_{lign}$ (D), `InstSim` (I) and `BALAS` (B).

	$F=4$					$F=6$					$F=8$				
	Q	M	D	I	B	Q	M	D	I	B	Q	M	D	I	B
$\bar{\eta}^0_{200}(\%)$	**47.0**	11.6	32.6	44.1	33.0	**52.5**	12.3	34.0	47.6	25.5	**54.9**	13.2	38.4	48.9	22.4
$\bar{\eta}^1_{200}(\%)$	**54.9**	13.2	38.4	48.9	22.4	**57.9**	18.2	44.7	51.9	22.4	**60.2**	23.6	48.7	54.2	23.4
$\bar{\eta}^2_{200}(\%)$	**57.9**	18.2	44.7	51.9	22.4	**60.9**	27.2	50.6	55.4	24.0	**63.1**	36.7	53.2	59.0	27.6
$\bar{\eta}^3_{200}(\%)$	**60.2**	23.6	48.7	54.2	23.4	**63.1**	36.7	53.2	59.0	27.6	**65.2**	47.8	55.4	61.6	31.1
$\bar{\eta}^4_{200}(\%)$	**61.5**	30.5	51.1	56.8	25.2	**64.6**	45.6	54.8	60.7	30.5	**67.1**	55.5	55.2	64.2	34.4
$\bar{\eta}^5_{200}(\%)$	**63.1**	36.7	53.2	59.0	27.6	**66.4**	52.4	55.6	63.1	32.4	**69.0**	60.2	54.9	66.7	38.1

Here *geometrical* means the results obtained after averaging across all the image classes. Note that the performance at the 0th round corresponds to the performance before starting relevance feedback, that is, when only the query image is used.

Figure 2 shows that with the increase of the rounds of relevance feedback and the number of images given in each round, the retrieval performance of all the compared methods improve. In particular, the improvement of `MindReader` is very significant. This is not difficult to understand since increasing either the rounds of relevance feedback or the number of images given in each round will result in the increase of the number of labeled examples, which is beneficial to the improvement of the retrieval performance. Specifically, since `MindReader` involves the estimation of the *ideal query* and a distance matrix [3], more labeled examples will lead to significantly better performance. Nevertheless, Figure 2 shows that the performance of `Qsim` is superior to `MindReader` and other compared methods. Moreover, it is impressive that no matter how many rounds of relevance feedbacks are executed and how many images are labeled in each round of the relevance feedback process, `Qsim` always achieves the best effectiveness, as shown in Table 1. These observations validate that `Qsim` is effective for attaining good performance in CBIR.

Considering that CBIR is an online task since the concept being queried can hardly be modelled before the user poses the query, time cost is an important factor of techniques in this area. So, the geometrical time costs of the compared methods spent in dealing with the initial query and each round of relevance feedback are compared, as shown in Table 2 where the smallest time cost at each round has been boldfaced. It can be found that `InstSim` is with the best efficiency. Although the time costs of `Qsim` are bigger than that of `InstSim`, `Qsim` is over 3 times faster than `MindReader` and `BALAS`, and over 4,780 times faster than $\mathtt{DA}_{lign}$. Therefore, it is evident that considering both the retrieval performance and time cost, `Qsim` is a good choice since it obtains the best retrieval performance with relatively small time cost.

4 Concluding Remarks

The implication of an image is often ambiguous, i.e., an image can usually be perceived with different meanings. So, the similarity between a pair of images should depend on not only the visual features of the images, but also the intention of the user, i.e., the concept being queried. However, previous image similarity measures are usually query-insensitive, which produce a static and constant similarity score for a pair of images despite which query is posed.

Table 2. Geometrical time costs (seconds) of `Qsim` (Q), `MindReader` (M), DA_{lign} (D), `InstSim` (I) and BALAS (B) spent in dealing with different rounds of relevance feedback.

	$F=4$					$F=6$					$F=8$				
	Q	M	D	I	B	Q	M	D	I	B	Q	M	D	I	B
round0	.0056	.0993	.3908	**.0028**	.0773	.0072	.1016	.2748	**.0037**	.0789	.0107	.0973	.3096	**.0034**	.0827
round1	.0107	.0973	.3096	**.0034**	.0827	.0155	.1055	.8876	**.0037**	.0922	.0211	.1009	2.9062	**.0040**	.1015
round2	.0155	.1055	.8876	**.0037**	.0922	.0238	.1010	4.4280	**.0041**	.1043	.0372	.0972	26.468	**.0048**	.1168
round3	.0211	.1009	2.9062	**.0040**	.1015	.0372	.0972	26.468	**.0048**	.1168	.0592	.0999	188.07	**.0051**	.1335
round4	.0286	.0976	11.006	**.0041**	.1086	.0531	.1008	106.58	**.0045**	.1289	.0873	.0952	829.78	**.0061**	.1503
round5	.0372	.0972	26.468	**.0048**	.1168	.0726	.0979	411.74	**.0064**	.1429	.1234	.0940	2805.8	**.0071**	.1663
average	.0198	.0996	6.9948	**.0038**	.0965	.0349	.1007	91.730	**.0045**	.1107	.0565	.0974	642.22	**.0051**	.1252

This paper advocates that query-sensitive similarity measures should be used in CBIR, which take the concept being queried into account. To illustrate that designing effective and efficient query-sensitive similarity measures is feasible, this paper proposes `Qsim`. By exploiting the query image as well as the images labeled by user in the relevance feedback process, `Qsim` can regularize the image similarities to some extent according to the concept being queried. Moreover, considering that relevant images usually belong to the same concept class but irrelevant images often belong to different concept classes, `Qsim` exploits relevant and irrelevant images differently. Experiments show that `Qsim` is superior to many state-of-the-art techniques, which validates the claim of the paper that query-sensitive similarity measures are superior to query-insensitive similarity measures in CBIR.

Due to the page limit, this paper does not present the comparison on `Qsim` and its degenerated variants, and the study on the influence of the choice of base similarity measure on the performance of `Qsim`. These results will be presented in a longer version. Since this is the first attempt of using query-sensitive similarity measure in CBIR, this paper has not tried to explore other kinds of query-sensitive similarity measures, and has not considered complicated settings such as numerical relevance, regional relevance, multi-concept query, user mistakes, etc. It is evident that designing other kinds of query-sensitive similarity measures is an interesting issue for future work. Applying `Qsim` to large image collections has also been left as a future issue.

Acknowledgments: Supported by FANEDD (200343), NSFC (60473046, 60505013), JiangsuSF (BK2005412).

References

[1] G. Ciocca and R. Schettini. A relevance feedback mechanism for content-based image retrieval. *Information Processing and Management*, 35(5):605–632, 1999.

[2] G. Giacinto and F. Roli. Instance-based relevance feedback for image retrieval. In L. K. Saul, Y. Weiss, and L. Bottou, editors, *Advances in Neural Information Processing Systems 17*, pages 489–496. MIT Press, Cambridge, MA, 2005.

[3] Y. Ishikawa, R. Subramanya, and C. Faloutsos. MindReader: Query databases through multiple examples. In *Proceedings of the 24th International Conference on Very Large Data Bases*, pages 218–227, New York, NY, 1998.

[4] 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 Systems for Video Technology*, 8(5):644–655, 1998.

[5] A. W. M. Smeulders, M. Worring, S. Santini, A. Gupta, and R. Jain. Content-based image retrieval at the end of the early years. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(12):1349–1380, 2000.

[6] A. Tombros and C. J. van Rijsbergen. Query-sensitive similarity measures for information retrieval. *Knowledge and Information Systems*, 6(5):617–642, 2004.

[7] N. Vasconcelos and A. Lippman. A unifying view of image similarity. In *Proceedings of the 15th International Conference on Pattern Recognition*, pages 1038–1041, Barcelona, Spain, 2000.

[8] G. Wu, E. Y. Chang, and N. Panda. Formulating content-dependent similarity functions. In *Proceedings of the 13th ACM International Conference on Multimedia*, pages 725–734, Singapore, 2005.

[9] R. Zhang and Z. Zhang. Stretching Bayesian learning in the relevance feedback of image retrieval. In *Proceedings of the 8th European Conference on Computer Vision*, pages 355–367, Prague, Czech, 2004.

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

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