27

FROM LOW LEVEL FEATURES TO HIGH LEVEL SEMANTICS

Cha Zhang and Tsuhan Chen

Department of Electrical and Computer Engineering

Carnegie Mellon University

Pittsburgh, Pennsylvania, USA

{ czhang, tsuhan }@andrew.cmu.edu

1. INTRODUCTION

A typical content-based information retrieval (CBIR) system, e.g., an image or video retrieval system, includes three major aspects: feature extraction, high dimensional indexing and system design [1]. Among the three aspects, high dimensional indexing is important for speed performance; system design is critical for appearance performance; and feature extraction is the key to accuracy performance. In this chapter, we will discuss various ways people have tried to increase the accuracy of retrieval systems.

If we think over what "accuracy" means for a retrieval system, we may find it very subjective and user-dependent. The similarity between objects can be very high-level, or *semantic*. This requires the system to measure the similarity in a way human being would perceive or recognize. Moreover, even given exactly the same inputs, different users probably have different feeling about their similarity. Therefore, retrieval system also needs to adapt to different users quickly through on-line user interaction and learning.

However, features we can extract from objects are often low-level features. We call these low-level features because most of them are extracted directly from digital representations of objects in the database and have little or nothing to do with human perception. Although many features have been designed for general or specific CBIR systems with high level concepts in mind, and some of them showed good retrieval performance, the gap between low-level features and high-level semantic meanings of the objects has been the major obstacle to better retrieval performance.

Various approaches have been proposed to improve the accuracy performance of CBIR systems. Essentially, these approaches fall into two main categories: to improve the features and to improve the similarity measures. Researchers have tried many features that are believed to be related with human perception, and they are still working on finding more. On the other hand, when the feature set is fixed, many algorithms have been proposed to measure the similarity in a way human beings might take. This includes off-line learning based on some training data, and on-line learning based on the user's feedback.

The chapter is organized as follows. Section 2 overviews some feature extraction algorithms that emphasize the high level semantics. Section 3 discusses the similarity measure. Section 4 presents some off-line learning methods for finding better similarity measures. Section 5 examines algorithms that learn on-line based on the user's feedback. Section 6 concludes the chapter.

2. EXTRACTING SEMANTIC FEATURES

Many features have been proposed for image/video retrieval. For images, often used features are color, shape, texture, color layout, etc. A comprehensive review can be found in [1]. Traditional video retrieval systems employ the same feature set on each frame, in addition to some temporal analysis, e.g., key shot detection [2][3][4][5]. Recently, a lot of new approaches have been introduced to improve the features. Some of them are based on temporal or spatial-temporal analysis, i.e., better ways to group the frames and select better key frames. This includes integrating with other media, e.g., audio, text, etc. Another hot research topic is motion-based features and object-based features. Compared to color, shape and texture, motion-based and object-based features are more natural to human being, and therefore at a higher level.

Traditional video analysis methods are often shot based. Shot detection methods can be classified into many categories, e.g., pixel based, statistics based, transform based, feature based and histogram based [6]. After the shot detection, key frames can be extracted in various ways [7]. Although key frames can be used directly for retrieval [3][8], many researchers are studying better organization of the video structures. In [9], Yeung *et al.* developed scene transition graphs (STG) to illustrate the scene flow of movies. Aigrain *et al.* proposed to use explicit models of video documents or rules related to editing techniques and film theory [10]. Statistical approaches such as Hidden Markov Model (HMM) [13], unsupervised clustering [14][15] were also proposed. When audio, text and some other accompanying contents are available, grouping can be done jointly [11][12]. There were also a lot of researches on extracting captions from video clips and they can also be used to help retrieval [16][17].

Motion is one of the most significant differences between video and images. Motion analysis has also been very popular for video retrieval. On one hand, motion can help find interesting objects in the video, such as the work by Courtney [18], Ferman *et al.* [19], Gelgon and Bouthemy [20], Ma and Zhang [21], etc. On the other hand, motion can be directly used as feature, named by Nelson and Polana [22] as "temporal texture". The work was extended by Otsuka *et al.* [23], Bouthemy and Fablet [24], Szummer and Picard [25], etc.

If objects can be segmented easily, object-based analysis of video sequence is definitely one of the most attractive methods to try. With the improvement on computer vision technologies, many object-based approaches were proposed recently. To name a few, in [18], Courtney developed a system, which allows for detecting moving objects in a closed environment based on motion detection. Zhong and Chang [26] applied color segmentation to separate images into homogeneous regions, and tracked them along time for content-based video query. Deng and Manjunath [27] proposed a new spatio-temporal segmentation and region-tracking scheme for video representation. Chang et al. proposed to use *Semantic Visual Templates (SVT)*, which is a personalized view of concepts composed of interactively created templates /objects.

This chapter is not intended to cover in depth for features used in the state-of-the-art video retrieval systems. Readers are referred to book Section II, III and V for more detailed information.

3. THE SIMILARITY MEASURE

Although new features are being discovered everyday, it is hard to imagine that we can find one set of features that can kill all the applications. Moreover, finding new features requires a lot of trials and errors, which in many cases has few clues to follow. If the feature set has been fixed for a certain retrieval system, another place that researchers can improve is the similarity measure.

It is said that a feature is good if and only if similar objects are close to each other in the feature space, and dissimilar objects are far apart. Obviously, to decide "close" or "far", the similarity measure plays an equally important role as the original feature space. For example, Euclidian distance or other Minkowski-type distances are widely used as similarity measures. A feature space is considered as “bad”, probably because it does not satisfy the above criterion under Euclidian distance. However, if the feature space can be “good” by first (nonlinearly) “warping” it and then applying the Euclidian distance or by employing some new distance metrics, it is still fine for our purpose. Here the “warping” can also be considered as a preprocessing step for the feature space. However, in this chapter, such kind of “warping” is included in similarity measure, thus gives the latter a fairly general definition.

The difficult problem is how to get the “warping” function, or, in general, find the right similarity measure. Santini and Jain suggested to do it based on psychological experiments [29]. They analysed some similarity measure proposed in the psychological literature to model human similarity perception, and showed that all of them actually challenge the Euclidean distance assumption in non-trivial ways. Their research implies that new distance metric is not only a preprocessing method, but also a necessity given the property of human perception. They developed a similarity measure, named *Fuzzy Feature Contrast*, based on fuzzy logic and Tverky’s famous *Feature Contrast* [30].

One problem with the psychological view is that it does not have sound mathematical or computational models. A much more popular approach to find the “warping” function is through *learning*. By learning we mean given a small amount of training data, the system can automatically find the right “warping” function to make the feature space better.

Before learning, one must decide what to use as the ground truth data, in other words, what one wants the “warping” function to be optimised for. It turns out that there are two major forms that are most often used: similarity/dissimilarity (SD) and keyword annotations (KA). If we know that some objects are similar to each other while others are not, the feature space should be “warped” so that similar objects get closer, and dissimilar objects get farther. If the training data has some keyword annotated for each object, we want objects that share the same keywords get closer while otherwise get farther. Both SD and KA have their advantages and good applications. SD is convenient for end-user, and it does not require any explanation why two objects are similar or not (sometimes the reason is hard to be presented to the system as an end-user). Therefore, SD is suitable for end-user optimised learning, e.g., to learn what the end-user really means by giving some examples. SD is almost exclusively used by relevance feedback – a very hot research topic today. KA is good for system maintainers to improve the general performance of the retrieval system. Recent work in video retrieval has shown an interesting shift from query by example (QBE) [31][1] to query by keywords (QBK). The reason is that it allows the end users to specify queries with keywords, as they have been used to in text retrieval. Moreover, KA allows the knowledge learned to be accumulated by simply adding more annotations, which is often not obvious when using SD. Therefore, by adding more and more annotations, the system maintainer can let the end-user feel that the system works better and better. Both SD and KA have their constraints, too. For example, SD is often too user-dependent and the knowledge obtained is hard to accumulate, while KA is often limited by a predefined small Lexicon.

The learning process is determined not only by the form of the training data, but also by their availability. Sometimes all the training data are available before the learning, and the process can be done in one batch. We often call this *off-line learning*. If the training data are obtained gradually and the learning is progressively refined, we call it *on-line learning*. Both cases were widely studied in the literature. In the following sections, we will focus on applying these learning algorithms on retrieval systems to improve the similarity measure. Off-line learning is discussed in Section 4 and on-line learning is in section 5.

4. OFF-LINE LEARNING

If all the training data is available at the very beginning, learning can be done in one step. This kind of off-line learning is often applied before the system is open to end-users. Although it might take quite some time, the speed is not a concern, as the end-user would not feel that.

Most off-line learning systems handle keyword annotations (KA). The keywords are often given as a predetermined set, organized in different ways. For example, Basu *et al.* [32] defined a Lexicon as relatively independent keywords describing events, scenes and objects. Many authors prefer the tree structure [34][35][33], as it is clean and easy to understand. Naphade *et al.* [36] and Lee *et al.* [37] used graph structure, which is appropriate if the relationship between keywords is very complex.

Once the training data is given, a couple of learning algorithms, parametric or non-parametric, can be used to learn the concepts behind the keywords. As far as the authors know, at least Gaussian Mixture Model (GMM) [32][35], Support Vector Machine (SVM) [38], Hybrid Neural Network [39], Multi-nets [36], Distance Learning Network [40] and Kernel Regression [33] have been studied in the literature. A common characteristic of these algorithms is that all of them can model potentially any distribution of the data. This is expected because we do not know how the objects that share the same concept are distributed in the low-level feature space. One assumption we can probably make is that in the low-level feature space, if two objects are very close to each other, they should be semantically similar, or be able to infer some knowledge to each other. On the other hand, if two objects are far from each other, the semantic link between them should be weak. Notice that because of the locality of the semantic inference, this assumption allows objects with the same semantic meaning to lie in different places in the feature space, which cannot be handled by simple methods such as linear feature reweighing. If the above assumption does not hold, probably none of the above learning algorithms will help improve the retrieval performance too much. The only solution to this circumstance might be to find better low-level features for the objects.

Different learning algorithms have different properties and are good for different circumstances. Take the Gaussian Mixture Model as an example. It assumes that the objects having the same semantic meaning are clustered into groups. The groups can lie at different places in the feature space, but each of them follows a Gaussian distribution. If the above assumptions are true, GMM is the best way to model the data: it is simple, elegant, easy to solve with algorithms such as EM [41][42] and sound in theoretical point of view. However, the above assumptions are very fragile: we do not know how many clusters the GMM will have, and no real case will happen that each cluster is a Gaussian Distribution. Despite the constraints, GMM is still very popular for its many advantages. Kernel regression (KR) is another popular machine learning technique. Instead of using a global model like GMM, KR assumes some local inference (kernel function) around each training sample. From the unannotated object's point of view, to predict its semantic meaning, an annotated object that is closer will have a higher influence, and a farther one will have less. Therefore, it will have similar semantic meanings to its close-by neighbours. KR can model any distribution naturally, and also has sound theory behind it [43]. The limitation of KR is that the kernel function is hard to select, and the number of samples needed to achieve a reasonable prediction is often high. Support Vector Machine (SVM) [44][45] is a recent addition to the toolbox of machine learning algorithms that has shown improved performance over standard techniques in many domains. It has been one of the most favourite methods among researchers today. The basic idea is to find the hyperplane that has the maximum *margin* towards the sample objects. *Margin* here means the distance the hyperplane can move along its normal before hitting any sample object. Intuitively, the greater the margin, the less the possibility that any sample points will be misclassified. For the same reason, if a sample object is far from the hyperplane, it is less likely to be misclassified. If the reader agree with the reasoning above, he/she will easily understand the SVM Active Learning approaches introduced in Section 5. For detailed information on SVM, please refer to [44][45].

Although after applying the learning algorithm, the semantic model can be used to tell the similarity between any two objects already, most systems require a fusion step [33][32][35]. The reason is that the performance of the statistically learned models is largely determined by the size of the training data set. Since often the training data is manually made, very expensive and thus small, it is risky to believe that the semantic model is good enough. In [33], semantic distance is combined with low-level feature distance through a weighting mechanism to give the final output, and the weight is determined by the confidence of the semantic distance. In [32], several GMM models are trained for each feature types, and the final result is generated by fusing the outputs of all the GMM models. In [35] where audio retrieval was studied, the semantic space and the feature space are designed symmetrically, i.e., each node in the semantic model is linked to equivalent sound documents in the acoustic space with a GMM, and each audio file/document is linked with a probability model in the semantic space. The spaces themselves are organized with hierarchical models. Given a new query in any space, the system can first search in that space to find the best node, and then apply the link model to get the retrieval results in the other space.

Keyword annotation is very expensive because it requires a lot of manual work. Chang and Li [46] proposed to employ another way of getting the ground truth data. They used 60,000 images as the original set and synthesize another set by 24 transforms such as rotation, scaling, cropping, etc. Obviously, images after the transforms should be similar to the one before the transform. They discovered a perceptual function called *dynamic partial distance function* (DPF). Synthesizing new images by transforms and using them as training data is not new. For example, people play this trick in face recognition systems when the training image set has very few images (e.g., only one). Despite the fact that transforms may not be complete as a model of similarity, this is a very convenient way of getting a lot of training data, and DPF seems to have reasonable performance as reported in [46].

5. ON-LINE LEARNING

Compared to off-line learning, on-line learning does not have the whole set of training data beforehand. The data are often obtained during the process, which makes the learning process a best effort one and highly dependent on the input training data, even the order they come in. However, on-line learning involves the interaction between the system and the user. The system can then quickly modify its internal model in order to output good results for each specific user. As discussed in Section 1, similarity measure in information retrieval systems is highly user-dependent. On-line learning's adaptive property makes it very suitable for such applications.

In retrieval systems, on-line learning is used in three scenarios: relevance feedback, finding the query seed, and enhancing the annotation efficiency. They are discussed respectively in the following three subsections.

5.1 RELEVANCE FEEDBACK

Widely used in text retrieval [47][48], relevance feedback was first proposed by Rui et al. as an interactive tool in content-based image retrieval [49]. Since then it has been proven to be a powerful tool and has become a major focus of research in this area [50][51][52][53][54][55]. Chapter 23 and Chapter 33 have detailed explanation on this topic.

Relevance feedback often does not accumulate the knowledge the system learned. That's because the end-user's feedback is often unpredictable, and inconsistent from user to user, or even query to query. If the user who gives the feedback is trustworthy and consistent, feedback can be accumulated and added to the knowledge of the system, as was suggested by Lee *et al.* [37].

5.2 QUERY CONCEPT LEARNER

In a query by example [31][1] system, it is often hard to initialise the first query, because the user may not have a good example to begin with. Having got used to text retrieval engines such as Google [56], users may prefer to query the database by keyword. Many systems with keyword annotations can provide such kind of service [32][33][35]. Chang *et al.* recently proposed the SVM Active Learning system [58] and MEGA system [57], which can be an alternate solution.

SVM Active Learning and MEGA have similar ideas but with different tools. They both want to find a query-concept learner that learns query criteria through an intelligent sampling process. No example is needed as the initial query. Instead of browsing the database completely randomly, these two systems ask the user to provide some feedback and try to quickly capture the concept in the user's mind. The key to success is to maximally utilize the user's feedback and quickly reduce the size of the space that the user's concept lies in. Active learning is THE answer.

Active learning is an interesting idea in the machine learning literature. While in traditional machine learning research, the learner typically works as a passive recipient of the data, active learning enables the learner to use its own ability to respond to collect data and to influence the world it is trying to understand. A standard passive learner can be think of as a student that sits and listens to a teacher, while an active learner is a student that asks the teacher questions, listens to the answers and asks further questions based on the answer. In the literature, active learning has shown very promising results in reducing the number of samples required to finish a certain task [59][60][61].

In practical, the idea of active learning can be translated into a simple rule: if the system is allowed to propose samples and get feedback, always propose those samples that the system is most confused of, or that can bring the greatest information gain.

Following the rule, SVM Active Learning becomes very straightforward. In SVM, objects far away from the separating hyperplane are easy to classify. The most confused objects are those that are close to the boundary. Therefore, during the feedback loop, the system will always propose the images closest to the SVM boundary for the user to annotate.

MEGA system models the query concept space (QCS) in *k*-CNF and the candidate concept space (CCS) in *k*-DNF [62]. Here *k*-CNF and *k*-DNF are Boolean formulae sets that can virtually model any practical query concepts. The CCS is initialised to be larger than the real concept space, while the QCS is initialised smaller. During the learning process, the QCS keeps being refined by the positive feedbacks, while the CCS keeps shrinking due to the negative samples. The spatial difference between QCS and CCS is the interesting area where most images are undetermined. Based on the idea of active learning, they should be shown to the user for more feedback. Some interesting trade-offs have to be made in selecting these samples [57].

5.3 EFFICIENT ANNOTATION THROUGH ACTIVE LEARNING

Keyword annotation is a very expensive work, as it can only be done manually. It is natural to look for methods that can improve the annotation efficiency. Active learning turns out to be also suitable for this job.

In [33], Zhang and Chen proposed a framework for active learning during the annotation. For each object in the database, they maintain a list of probabilities, each indicating the probability of this object having one of the attributes. During training, the learning algorithm samples objects in the database and presents them to the annotator to assign attributes to. For each sampled object, each probability is set to be one or zero depending on whether or not the corresponding attribute is assigned by the annotator. For objects that have not been annotated, the learning algorithm estimates their probabilities with biased kernel regression. Knowledge gain is then defined to determine, among the objects that have not been annotated, which one the system is the most uncertain of. The system then presents it as the next sample to the annotator to assign attributes to.

Naphade *et al.* proposed a very similar work in [38]. However, they used a support vector machine to learn the semantics. They have essentially the same method as Chang *et al.*'s SVM Active Learning [58] to choose new samples for the annotator to annotate.

6. CONCLUSION

In this chapter, we overviewed the various ways people use to improve the accuracy performance of content-based information retrieval system. The gap between low-level features and high level semantics has been the main obstacle for developing more successful retrieval systems. It can be expected that it will still remain as one of the most challenging research topic in this field.

REFERENCES

[1] Y. Rui and T. S. Huang, “Image Retrieval: Current Techniques, Promising, Directions and Open Issues”, *Journal of Visual Communication and Image Representation*, Vol. 10, No. 4, April 1999.

[2] A. Nagasaka and Y. Tanaka, “Automatic Video Indexing and Full-Video Search for Object Appearance”, Proc. Of IFIP 2nd Working Conf. On Visual Database Systems, pp. 113-127, 1992.

[3] H. J. Zhang, J. H. Wu, D. Zhong, and S. W. Smoliar, “Video Parsing, Retrieval and Browsing: An Integrated and Content-Based Solution”, *Pattern Recognition*, pp. 643-658, Vol. 30, No. 4, 1997.

[4] Y. Deng and B. S. Manjunath, “Content-Based Search of Video Using Color, Texture and Motion”, *Proc. Of IEEE Intl. Conf. On Image Processing*, pp. 534-537, Vol. 2, 1997.

[5] P. Aigrain, H. J. Zhang and D. Petkovic, “Content-Based Representation and Retrieval of Visual Media: A State-of-the-Art Review”, *Int. J. Multimedia Tools Appl.*, pp.179—202, Vol. 3, November 1996.

[6] J. S. Boresczky and L. A. Rowe, "A comparison of video shot boundary detection techniques," *Storage & Retrieval for Image and Video Databases IV, Proc. SPIE 2670*, pp. 170--179, 1996.

[7] W. Wolf, "Key frame selection by motion analysis", *IEEE ICASSP*, pp. 1228-1231, Vol. 2, 1996.

[8] M. Flickner *et al.*, “Query by Image and Video Content”, IEEE Computer, pp23-32, September 1995.

[9] M. Yeung, B. L. Yeo and B. Liu, “Extracting Story Units from Long Programs for Video Browsing and Navigation”, *Proc. IEEE Conf. On Multimedia Computing and Systems*, pp. 296-305, 1996.

[10] P. Aigrain, P. Joly, and V. Longueville. Medium knowledgebased macro-segmentation of video into sequences. In M. T. Maybury, editor, *Intelligent Multimedia Information Retrieval*, pp.159–173. AAAI/MIT Press, 1997.

[11] A. G. Hauptmann and M. A. Smith. “Text, Speech, and Vision for Video Segmentation: The Informedia Project.” In *AAAI Fall*

Symposium, Computational Models for Integrating Language and Vision, Boston, 1995.

[12] R. Lienhart, S. Pfeiffer, and W. Effelsberg. "Scene Determination Based on Video and Audio Features", Technical report, University of Mannheim, November 1998.

[13] J. S. Boreczky and L.D. Wilcox, "A hidden Markov model framework for video segmentation using audio and image features," *IEEE ICASSP*, pp. 3741-3744, Vol. 6, Seattle, 1998.

[14] Y. Rui, T. S. Hunag, and S. Mehrotra. "Constructing Table-of-Content for Videos", *ACM Multimedia Systems Journal, Special Issue Multimedia Systems on Video Libraries*, pp.359-368, Vol. 7, No. 5, Sep. 1999.

[15] D. Zhong, H. Zhang, S.-F. Chang: "Clustering Methods for Video Browsing and Annotation", *SPIE Conference on Storage and Retrieval for Image and Video Databases*, pp.239-246, Vol. 2670, 1996.

[16] U. Gargi, S. Antani and R. Kasturi, "Indexing Text Events in Digital Video Databases", Porc. 14th Int'l Conf. Pattern Recognition, pp. 916-918,1998.

[17] J.C. Shim, C. Dorai, and R. Bolle, "Automatic Text Extraction from Video for Content-Based Annotation and Retrieval", Proc. 14th Int'l Conf. Pattern Recognition, pp. 618-620, 1998.

[18] J. D. Courtney, "Automatic Video Indexing via Object Motion Analysis", Pattern Recognition", pp. 607-625, Vol. 30, No. 4, 1997.

[19] A. M. Ferman, A. M. Tekalp, and R. Mehrotra, "Effective Content Representation for Video", *Proc. of ICIP'98*, pp. 521-525, Vol. 3, 1998.

[20] M. Gelgon and P. Bouthemy. "Determining a Structured Spatio-Temporal Representation of Video Content for Efficient Visualization and Indexing." *Proc. 5th Eur. Conf. on Computer Vision, ECCV'98*, Freiburg, June 1998.

[21] Y. F. Ma and H. J. Zhang, "Detecting Motion Object by Spatio-Temporal Entropy", IEEE International Conference on Multimedia and Expo, Tokyo, Japan, August 22-25, 2001.

[22] R. C. Nelson and R. Polana, ``Qualitative Recognition of Motion Using Temporal Texture'', *Proc. DARPA Image Understanding Workshop*, San Diego, CA, pp.555-559, Jan. 1992.

[23] K. Otsuka, T. Horikoshi, S. Suzuki and M. Fujii, "Feature Extraction of Temporal Texture Based on Spatio-Temporal Motion Trajectory", *Proc. 14th Int. Conf. On Pattern Recognition, ICPR'98*, pp.1047-1051, Aug. 1998.

[24] P. Bouthemy, R. Fablet. "Motion Characterization from Temporal Cooccurrences of Local Motion-Based Measures for Video Indexing", *Int. Conf on Pattern Recognition, ICPR'98*, pp.905-908, Vol. 1, Australia, Aug. 1998.

[25] M. Szummer and R. W. Picard, “Temporal Texture Modeling”, *IEEE ICIP’96*, pp.823-826, Sep. 1996.

[26] D. Zhong and S.F. Chang, "Video Object Model and Segmentation for Content-Based Video Indexing," *IEEE International Symposium on Circuits and Systems*, Hong Kong, June 1997.

[27] Yining Deng and B. S. Manjunath, “Netra-V: Toward an Object-Based Video Representation”, IEEE Trans. CSVT, pp.616-627, Vol. 8, No. 3, 1998.

[28] S. F. Chang, W. Chen and H. Sundaram, “Semantic Visual Templates: Linking Visual Features to Semantics”, *IEEE ICIP*, 1998.

[29] S. Santini, and R. Jain, "Similarity Measures", *IEEE Trans. Pattern Analysis and Machine Intelligence*, pp.871-883, Vol. 21, No. 9, Sep. 1999.

[30] A. Tversky, “Features of Similarity”, *Psychological Review*, pp.327-352, Vol. 84, No. 4, July 1977.

[31] M. Flicker, H. Sawhney, W. Niblack, J. Ashley, Q. Huang, B. Dom, M. Gorkani, J. Hafner, D. Lee, D. Petkovic, D. Steele and P. Yanker, “Query by Image and Video Content: The QBIC System”, *IEEE Computer*, pp. 23-32, Vol. 28, No. 9, 1995.

[32] S. Basu, M. Naphade and J. R. Smith, “A Statical Modeling Approach to Content Based Retrieval”, *IEEE ICASSP*, 2002.

[33] C. Zhang and T. Chen, "An Active Learning Framework for Content Based Information Retrieval", *IEEE trans. on Multimedia, Special Issue on Multimedia Database*, pp. 260-268, Vol. 4, No. 2, June 2002.

[34] Y. Park, “Efficient Tools for Power Annotation of Visual Contents: A Lexicographical Approach”, *ACM Multimedia*, pp. 426-428, 2000.

[35] M. Slaney, “Semantic-Audio Retrieval”, *IEEE ICASSP*, 2002.

[36] M. R. Naphade, I. Kozintsev, T. S. Huang and K. Ramchandran, “A Factor Graph Framework for Semantic Indexing and Retrieval in Video”, *Proceedings of the IEEE Workshop on Content-based Access of Image and Video Libraries*, 2000.

[37] C. S. Lee, W.-Y. Ma and H. J. Zhang, “Information Embedding Based on User’s Relevance Feedback for Image Retrieval”, *Invited paper, SPIE Int. Conf. Multimedia Storage and Archiving Systems IV,* Boston, pp.19-22, Sep. 1999.

[38] M. R. Naphade, C. Y. Lin, J. R. Smith, B. Tseng and S. Basu, “Learning to Annotate Video Databases”, *SPIE Conference on Storage and Retrieval on Media databases*, 2002.

[39] W. Y. Ma and B. S. Manjunath, “Texture Features and Learning Similarity”, *IEEE Proceedings CVPR '96*, pp. 425-430, 1996.

[40] D. McG. Squire, “Learning a Similarity-Based Distance Measure for Image Database Organization from Human Partitionings of an Image

Set", *IEEE Workshop on Applications of Computer Vision (WACV'98)*, pp.88-93, 1998.

[41] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum-likelihood from Incomplete Data via the EM Algorithm", *J. Royal Statist. Soc., Ser. B*, pp. 1-38, Vol. 39, No. 1, 1977.

[42] G. McLachlan, and T. Krishnan, *The EM algorithm and Extensions*. Wiley series in probability and statistics. John Wiley & Sons. (1997).

[43] R. O. Duda, P. E. Hart and D. G. Stork, *Pattern Classification (2nd Edition)*, John Wiley & Sons, New York, 2000.

[44] C. Burges, "A Tutorial on Support Vector Machines for Pattern Recognition", Data Mining and Knowledge Discovery, pp.121-167, Vol. 2, No. 2, 1998.

[45] N. Cristianini and J. Shawe-Taylor, *An Introduction to Support Vector Machines and Other Kernel-Based Learning Methods*, Cambridge University Press, 2000.

[46] E. Chang and B. Li, "On Learning Perceptual Distance Function for Image Retrieval", *IEEE ICASSP*, 2002.

[47] Donna Harman, "Relevance Feedback Revisited", *Proceedings of the Fifteenth Annual International ACM SIGIR conference on Research and development in information retrieval*, pp. 1-10, 1992.

[48] Gerald Salton and Chris Buckley, "Improving Retrieval Performance by Relevance Feedback", *Journal of the American Society for Information Science*, pp. 288-297, Vol. 41, No. 4, 1990.

[49] Yong Rui, Thomas S. Huang, Michael Ortega, and Sharad Mehrotra, "Relevance Feedback: A Power Tool for Interactive Content-based Image Retrieval", *IEEE Trans. On Circuits and Systems for Video Technology*, pp. 644-655, Vol. 8, No. 5, Sep. 1998.

[50] Y. Ishikawa, R. Subramanya, and C. Faloutsos, "Mindreader: Query Database through Multiple Examples", *Proceeding of the 24th VLDB Conference*, New York, 1998.

[51] Yong Rui and Thomas S. Huang, "Optimizing Learning in Image Retrieval", *Proceeding of IEEE int. Conf. On Computer Vision and Pattern Recognition*, Jun. 2000.

[52] Qi Tian, Pengyu Hong, Thomas S. Huang, "Update Relevant Image Weights for Content-based Image Retrieval Using Support Vector Machines", *Multimedia and Expo, 2000. ICME 2000. 2000 IEEE International Conference on*, pp. 1199-1202, Vol. 2 , 2000.

[53] Sanghoon Sull, Jeongtaek Oh, Sangwook Oh, S. Moon-Ho Song, Sang W. Lee, "Relevance Graph-based Image Retrieval", *Multimedia and Expo, 2000. ICME 2000. 2000 IEEE International Conference on*, pp. 713 –716, vol. 2, 2000.

[54] Nikolaos D. Doulamis, Anastasios D. Doulamis and Stefanos D. Kollias, "Non-linear Relevance Feedback: Improving the Performance of Content-based Retrieval Systems", *Multimedia and Expo, 2000.*

ICME 2000. 2000 IEEE International Conference on, pp. 331-334, Vol. 1, 2000.

[55] T. P. Minka and R. W. Picard, "Interactive Learning Using a 'Society of Models'", *M. I. T Media Laboratory Perceptual Computing Section Technical Report* No. 349.

[56] www.google.com

[57] E. Chang and B. Li, "MEGA -- The Maximizing Expected Generalization Algorithm for Learning Complex Query Concepts", *UCSB Technical Report*, August 2001.

[58] S. Tong and E. Chang, "Support Vector Machine Active Learning for Image Retrieval", *ACM Multimedia*, 2001.

[59] David A. Cohn, Zoubin Ghahramani, and Michael I. Jordan, "Active Learning with Statistical Models", *Journal of Artificial Intelligence Research*, pp. 129-145, 4, 1996.

[60] A. Krogh and J. Vedelsby, "Neural network Ensembles, Cross Validation, and Active Learning", In G. Tesauro, D. Touretzky, and T. Leen, editors, *Advances in Neural Information Processing Systems*, volume 7, Cambridge, MA, 1995. MIT Press.

[61] David D. Lewis and William A. Gale, "A Sequential Algorithm for Training Text Classifiers", *ACM-SIGIR 94*, pp. 3-12, Springer-verlag, London, 1994.

[62] M. Kearns, M. Li and L. Valiant, "Learning Boolean Formulae", Journal of ACM, pp.1298-1328, Vol. 41, No. 6, 1994.

[63]

INDEX