Beyond Recall and Precision: A Full Framework for MIR System Evaluation

Josh Reiss
Department of Electronic Engineering
Queen Mary, University of London
Mile End Road,
London E1 4NS UK
+44-207-882-5528
josh.reiss@elec.qmul.ac.uk

Mark Sandler
Department of Electronic Engineering
Queen Mary, University of London
Mile End Road,
London E1 4NS UK
+44-207-882-7680
mark.sandler@elec.qmul.ac.uk

ABSTRACT

The Cranfield tests are perhaps the most well-known and often cited example of benchmarking of information retrieval systems. However, of the six criteria that Cleverdon identified as pertinent for analysis of information retrieval systems, only two, precision and recall, are typically investigated. We argue that the other criteria are also vitally important for advanced IR systems such as a music information retrieval (MIR) system. They should be modified and put into the appropriate framework for MIR systems. Furthermore, a systematic method of measuring all valid criteria should be devised. This paper considers similar attempts with other advanced IR systems, and suggests how to establish and measure the appropriate criteria for information retrieval systems to be used in conjunction with a Music Digital Library (MDL).

1. INTRODUCTION

In 1946 Cyril Cleverdon, the librarian of Cranfield College, embarked on a series of major research projects that have since become known as the Cranfield Experiments.[1, 2] This work was one of the most important contributions that shaped the field of information science in the 1950s and 60s. Cleverdon, Mills and Keen[3] analysed the measurable factors that are to be taken into consideration for the appraisal of an IR system:

1. The *coverage* of the collection: the extent to which the system includes relevant matter.

2. The *time* lag: the average interval between the time the request is made and the time an answer is given.

3. The form of *presentation* of the output.

4. The *effort* involved on the part of the user to obtain answers to his search requests.

5. The *recall* of the system: the proportion of relevant material that is actually retrieved in answer to a search request.

6. The *precision* of the system: the proportion of retrieved material that is actually relevant.

There has been a significant body of work that has considered the meaning of recall, precision and relevance. Numerous alternative definitions have been suggested[4, 5], and precision and recall have also found application in linguistics[6] and document analysis[7], among others. However, the other four measurable quantities, coverage, time, presentation and effort, are often ignored. Indeed, some authors have quickly dismissed their importance, stating that they can be readily accessed and thus further discussion is unnecessary.[8]

Although this may be true in certain cases, for instance, in regards to evaluation of text information retrieval systems on the same testbed, it is not necessarily true in regards to the MIR/MDL systems that are considered here. In this paper, we will discuss in detail important measurable quantities other than precision and recall that are pertinent to the evaluation of MIR systems.

2. CHOICE OF EVALUATION CRITERIA

First, we note that in the benchmarking of IR systems, these six measurable factors have been modified to make them more specific to the information retrieval system under consideration. One example is given from the benchmarking of World Wide Web search engines. Chu and Rosenthal[9] proposed the following five factors be used in an evaluation methodology for WWW search engine.

1. *Composition* of Web indexes – This incorporates coverage, update frequency and index method..

2. Search *capability* – This measures the inclusion of various useful and common features in ability to make a search request. This includes Boolean logic, phrase searching, truncation, and limiting facilities.

3. Retrieval *performance*– This incorporates precision, recall, and response time.

4. Output option (*presentation*) – This measures both the number of output options that are available and the actual content of those options.

5. User *effort* – This refers to documentation and interface.

The important points to make from considering Chu and Rosenthal's proposed factors is that they recognized that the criteria are very much problem specific. In a sense, all the web search engines used the same testbed (the entire World Wide Web), but differed widely on how they indexed that testbed. Search capability is also not considered in many IR evaluation studies, in part because often only systems with the same capabilities are compared.

Similarly, measurement of retrieval performance tells nothing regarding the content of the output. This output content is not fully incorporated into the presentation criterion suggested by Cleverdon, but it is an important factor in MIR systems. For instance, the ability to listen to a small sample of the retrieved musical pieces without having to download large files is a very useful feature that would be measured in the output options.

Cleverdon's criteria could also be modified to make it more specific to MIR systems and to make it more relevant to the current state of the IR field. Effort and presentation both are effected by user interface design. Thus UI design could be considered a separate criterion. Time lag is highly dependent on the nature of the testbed, and thus time lag could be replaced by computational measures that isolate speed issues that are not related to coverage. However, in order to put the following discussion in a familiar framework, the authors have chosen not to deviate from Cleverdon's six criteria. Instead, we concentrate on how each of the criteria are important in MIR system evaluation, and how they should be quantified and adapted to musical queries.

3. COVERAGE

The issues regarding the size and format of the testbed are dealt with in a variety of other papers in this and related sessions ([10] and references therein). Choice of appropriate coverage is fundamentally dependent on how the data collection is to be used and who will use it. Much of the work in the field of MIR has been conducted by musicologists, but many possible applications exist that are relevant to anyone with an interest in music. Thus both the goals of the musicologists and those of the larger audience and should be incorporated. The simplest suggestion therefore is to incorporate as much music as possible, for as many genres as possible. It is the responsible of the MIR system, not the library, to effectively index this system and to effectively retrieve relevant documents for a given query.

Further specific requirements for an evaluation testbed were given in [11]. We reiterate here that an evaluation testbed should include records most pertinent for evaluation queries. This goes beyond queries that are used in precision and recall evaluation. Records that might yield exceptionally long time lags, make presentation difficult, or require significant effort for retrieval must be included in the testbed so that these criteria may be evaluated.

Other issues regarding coverage are discussed in the section devoted to the presentation of the output.

4. TIME LAG

Most studies of IR systems dismiss timing information using the reasoning that it is both easily assessed and that sufficient hardware resources will make the time lag sufficiently small as to be unimportant.[1]

There are several flaws here. First and foremost is that timing information is *not*, in general, easily assessed. It depends on a multitude of factors, including the nature and length of the query, the size of the collection, the indexing scheme used, the hardware and software implementation, and the amount of network or internet traffic.

One may wish to ignore traffic issues as being beyond control, and ignore implementation issues using the assumption that in a fair comparison, two different MIR systems would share the same network, software design and hardware. This assumption is not valid, though, in situations where the searching and indexing might be traffic dependent (e.g., peer-to-peer gnutella based networks such as www.AudioFind.com or www.mp3Board.com). In effect, the collection size is increased in conjunction with an increase in time lag. In such a situation, designers attempt to achieve a fair balance between time lag over the network, and recall and precision. Thus the software and hardware choices are also important factors. In addition, no amount of hardware will fully alleviate the time lag problems, since the lag is primarily caused by network traffic and bandwidth issues, not by hardware based computation.

Music is fundamentally multidimensional. With the exception of monophonic music, at any given time several notes may represent what is occurring in the music. Information such as timbre, duration, and loudness may also be incorporated as further dimensions inherent in the data. Even when abstract feature extraction is used to represent audio data, several features such as frequency, intensity, frequency envelope, are required in order to accurately describe a short sample of audio. Whether feature extraction, transcription or a straightforward search of encoded metadata is used, a search represents a proximity or exact match search on multidimensional data.

[1] See, for example,

www.scism.sbu.ac.uk/inmandw/tutorials/irtutorials/H1.DOC

or pi0959.kub.nl/Paai/Onderw/V-I/Content/evaluation.html

This is not a trivial problem. One dimensional data can be easily sorted, and thus indexing schemes on one dimensional data such as text are relatively straightforward. But with true multidimensional searches, where one wants proximity in several dimensions, the appropriate sorting and searching method is not clear. A linear search would require investigating all records in the database for the closest match. For *n* records, this is an Order(*n*) problem. In the case of one dimensional data, this can be reduced to an O(log *n*) problem in most situations. Hardware improvements in general can only speed up the search by a constant amount. Thus no amount of hardware will reduce the complexity of the search, and an efficient indexing scheme is required.

There is a body of work in computer science related to efficient multidimensional searches. Some of this was discussed and applied to MIR in [12]. The various search structures all deal with how to optimize searching of multidimensional data through the creation of an appropriate index. The constraints typically include optimization for certain query types, data structures, testbed size, or dimensionality. Some of the simplest structures do not involve the creation of a search tree, but instead just order the data based on the multidimensional structure.[12-14] However, these structures are often very limited in their efficiency, and only outperform more advanced tree-based indexes in the case of low dimensionality and/or certain specific data distributions.

A simple example of how a multidimensional search tree may make a search more efficient is depicted in Figure 1. Consider the hypothetical situation where only two features have been extracted from audio data. For a given window in an audio piece, the dominant frequency (roughly corresponding to pitch) and the duration of that frequency (corresponding to note duration) are given. Thus a query asking for a similar sound would look for a nearest neighbor in the data to the given two dimensional data point. One index structure that could be used here is the kd-tree[15], which sorts the data by making cuts in a given dimension and sorting points into values below or above the cut. The tree is optimized and balanced by choosing each cut based upon the distribution of the data.

The figure demonstrates how most of the data need not be checked. A search descends down the tree by comparing the query's frequency and duration to the cuts used at nodes in the tree. One then ascends the tree to find a nearest neighbor. Typically, very few data points need actually be compared before it can be shown that all remaining data must be further away than the nearest neighbor found so far. Although not rigorously proven, it can be argued that in many situations this reduces a nearest neighbor search from order *n* time to order log *n* time.

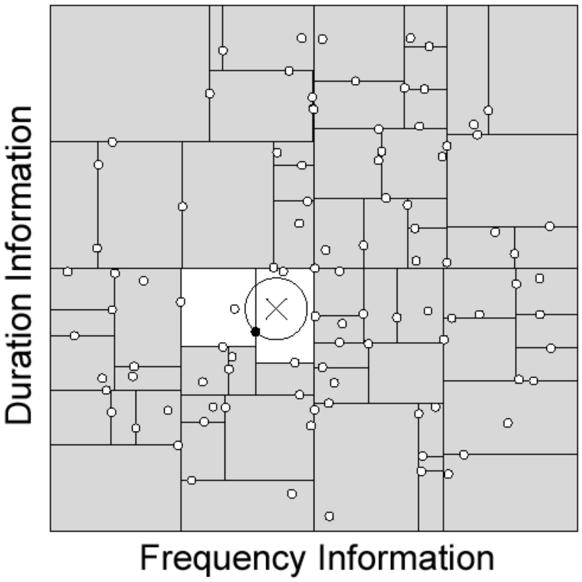


Figure 1. An example depicting an indexing of two dimensional data using a kd-tree. Two features, note duration and frequency have been extracted. Use of the kdtree helps minimize the number of records that need to be searched in order to find the closest record to the query. Adapted from a figure in [16]

However, feature extraction techniques often use a large number of features to describe music or musical segments. Feature extraction has thus become an ongoing research area, regardless of the application (see [17] and other articles in the same journal issue). Thus index structures are required which are more suitable to high dimensional data (i.e., more than six features are extracted for each musical segment). Vantage point trees have been shown to outperform the kd-tree in many high dimensional neighbor query problems[18]. The family of R-Tree based search structures[19, 20] are designed for higher dimensional problems with spatial data, and thus may be more applicable for situations when the features are not simply scalar values. If string matching techniques are used, such as in transcription based queries[21, 22] then index structures designed for multidimensional subsequences may be more appropriate.[23, 24]

Clearly, for sophisticated MIR systems that use large digital libraries, then speed and computation benchmarking becomes important. Here, we can use several measures from computer science that are more rigorous than simple time lag, which is more subject to unknowns. The authors propose the following measures as being relevant.

1. Estimation of computational order for nearest neighbor searches.

This can be simply computed by computing the average time lag for queries such as "Find me a piece of music that sounds like…" for a wide variety of musical queries over databases of various size. By estimating how the time lag between query and response varies as a function of database size, one should be able to estimate the order of complexity of the search algorithm, $O(n)$. It is important to note here that the constant factors in algorithmic complexities are often ignored and omitted, due to the assumption that for large databases, other terms will dominate. As demonstrated in [12], even for relatively large databases (approximately 100,000 records), constant factors may still yield a significant effect on computational differences between search methods.

2. Normalized CPU cost – The ratio of the average CPU time required to execute a query versus the average time to execute a query by using a brute force linear scan of the database.

This is related to measurement of computational order, but has several advantages. It is a more exact and meaningful measurement, since order is a theoretical value. Also, normalizing by the search time for a brute force search, and measuring only CPU time, eliminates factors such as network traffic, and minimizes the effects of processor speed and operating system used.

3. Normalised page access cost – The ratio of the number of disk accesses to execute a query versus the number of page accesses to execute a query by using a brute force linear scan of the database.

For large databases, files or metadata are not typically stored in main memory. Instead, they reside in secondary storage devices (magnetic disk hard drives) or even tertiary storage devices (removable media such as optical disks and tapes), as depicted in Figure 2. Disks for both secondary and tertiary storage are often organized into pages, and in magnetic disk storage systems access to sequential records on the same page is typically ten times faster than random access across pages. Thus advanced indexing techniques take advantage of this structure and organize the index such that the number of pages accessed in processing a query is minimized. Thus this measures how effectively the MIR system indexes the data in order to minimize access time. In most circumstances, the time lag is produced due to a combination of network transfer time, CPU time, and access time. Thus these measures effectively benchmark the speed of an MIR system.

5. PRESENTATION

The user interface of an IR system relates both to the ease of making a search request and the presentation of the results. But Cleverdon's criteria were given long before user interface design had become an established field of computer science. It is suggested therefore that one method by which to improve the presentation and minimize user effort in MIR systems, is through the development and evaluation of well-designed user interfaces.

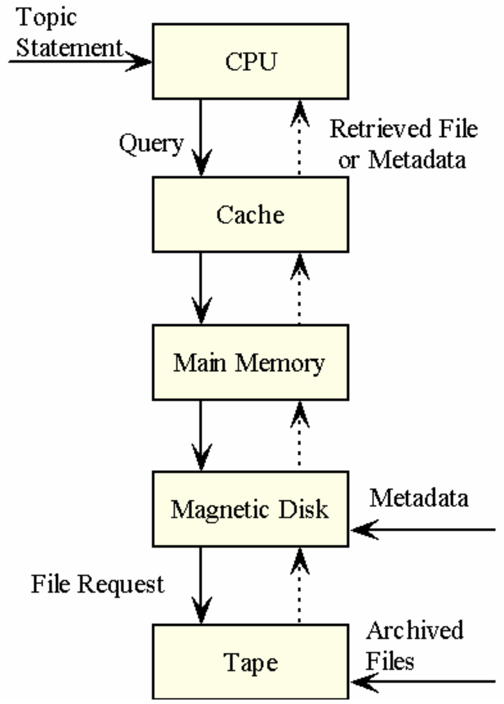


Figure 2. Processing an MIR query sent to a large MDL. Both organization of data in secondary storage and how the query method uses that organization are critical to speed and performance.

An excellent review of methods of presentation for modern information retrieval systems is provided in [25]. Presentation in existing MIR systems was evaluated in [11]. One important point to note is that relationships between relevant musical files may exist on a multitude of levels. These relationships should be taken into account in the presentation, and sometimes should be explained in the presentation and may even dictate which records to present.

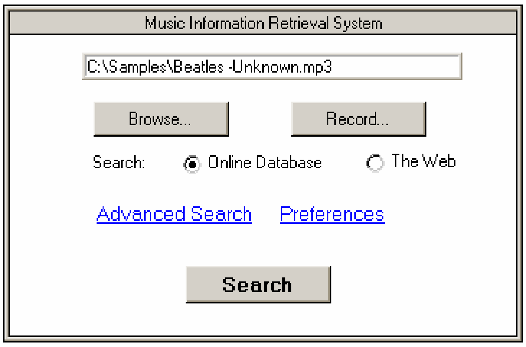
For instance, retrieved documents may include multiple performances of the same piece of music, and in multiple formats. As an extreme example, when one searches for music similar to the Beatles "Hey Jude," one does not in general want all live performances, or all the various quality mp3 compressed versions of the album track. This may be considered an aspect of the effort issue, and to some extent of the coverage, but one solution is found in terms of presentation of results.

Suppose instead that a record in the collection is simply considered to be the metadata that describes all versions of The Beatles "Hey Jude." When this record is retrieved, the results do not present each variation as a different record. Instead, the user may then narrow his search to consider just live versions, just uncompressed studio versions, etc. The metadata should thus incorporate knowledge of all versions available and how they are related, and this information should be appropriately presented in the results. This is analogous to when web search engines choose only to provide the most relevant document from a website, and offer the user the option of also presenting the other relevant documents.

6. EFFORT

In terms of effort, a compromise is sought between functionality and simplicity. One wishes to give the novice user the opportunity to quickly formulate his query, while at the same time giving him the ability to phrase a complex or highly specific query.

An example user interface is given in Figure 3. The only options concern how the query file is found and what collection is searched. The topic statement is simply "Find me music that sounds like …," where the query specifies either a previously stored file (e.g., on a network, the local hard drive, a web site, an ftp site, etc…) or one created for the query, (hummed, whistled or sung, played from the CD drive or through streaming audio, etc…). Searches may be performed on a database or across the internet (such as through peer-to-peer networks), and these two searches are distinguished because the results and timing are fundamentally different.



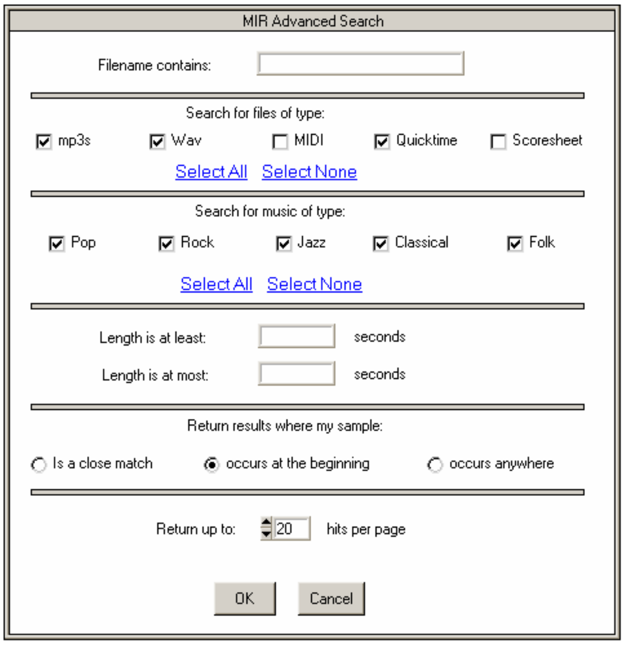
Music Information Retrieval System

C:\Samples\Beatles -Unknown.mp3

Search: ● Online Database ○ The Web

Figure 3. The front end user interface for a possible MIR system. Here, the user interface is kept as simple and intuitive as possible in order to facilitate use by novice users.

Figure 4 provides an example of what the advanced preferences might be. Again, they are not specific to the musicologist, but instead allow anyone to refine their search without knowledge of the internal complexities in the MIR system. In Figure 4, the authors attempted to present what might be popular options for the user to fine-tune his search. The user might have knowledge of the author or title, or perhaps know that he wants to retrieve the entire song, and not a popular sample. Thus options are given for specifying filenames and duration. Here, it was assumed that the files in the collection may have incorrect metadata or consist of a wide variety of formats, as is the case with WWW searches. An attempt was also made to incorporate many of the features that are specific to certain MIR tasks, such as genre classification ("Search for music of type") and optical recognition/transcription ("Search for files of type- Scoresheet").



MIR Advanced Search

Filename contains:

Search for files of type:

☑ mp3s ☑ Wav ☐ MIDI ☑ Quicktime ☐ Scoresheet

Search for music of type:

☑ Pop ☑ Rock ☑ Jazz ☑ Classical ☑ Folk

Length is at least: seconds

Length is at most: seconds

Return results where my sample:

○ Is a close match ● occurs at the beginning ○ occurs anywhere

Return up to: 20 hits per page

Figure 4. Advanced search options for a hypothetical MIR System. This window is displayed only if the user wants to refine his search parameters, but it should provide as many pertinent options as possible.

7. CONCLUSION

When one considers evaluation of an information retrieval system, one mainly is concerned with the ability of the system to accurately recall the most relevant documents and to be precise by omitting irrelevant documents. However, recall and precision are not sufficient when one wishes to evaluate the overall performance of such a system. Coverage, time lag, presentation and effort all affect the quality of an information retrieval system. For multimedia systems, time lags may be unacceptably long, and presentation is not obvious. MIR systems are still in their infancy. Design of an appropriate testbed and other coverage questions are yet to be clearly determined. Furthermore, the effort required by the user has yet to be determined because full MIR systems have not yet been developed.

Thus the author's believe that importance must be placed on *all* aspects of an MIR system in its evaluation and benchmarking. A simple benchmarking of existing online MIR systems was provided in [11]. We have attempted to provide a framework for benchmarking of future MIR systems. As suitable MDL testbeds are developed and MIR systems become more advanced, it is hoped that these suggestions are incorporated into a formal procedure for MIR/MDL system evaluation.

8. REFERENCES

1. Keen, M., *Cyril W. Cleverdon: bibliography.* Journal of Documentation, 1998. **54**(3): p. 274-280.
2. Keen, M., *Cyril W. Cleverdon.* Journal of Documentation, 1998. **54**(3): p. 265-273.
3. Cleverdon, C.W., J. Mills, and M. Keen, *Factors Determining the Performance of Indexing Systems, Volume I - Design, Volume II - Test Results, ASLIB Cranfield Project*, in *Readings in Information Retrieval*, K. Sparck Jones and P. Willett, Editors. 1997, Morgan Kaufmann: San Francisco.
4. Hersh, W.R., J. Pentecost, and D.H. Hickam, *A task-oriented approach to information retrieval evaluation.* Journal of the American Society for Information Science, 1996. **47**: p. 50-56.
5. Tague-Sutcliffe, J.M., *Some Perspectives on the Evaluation of Information Retrieval.* Journal of the American Society for Information Science, 1996. **47**(1): p. 1-3.
6. Radev, D.R. *Learning correlations between linguistic indicators and semantic constraints: Reuse of context-dependent descriptions of entities.* in *The Joint 17th International Conference on Computational Linguistics 36th Annual Meeting of the Association for Computational Linguistics (COLING-ACL'98)*. 1998. Montreal, Canada.
7. Junker, M., R. Hoch, and A. Dengel. *On the Evaluation of Document Analysis Components by Recall, Precision, and Accuracy.* in *ICDAR 99, Fifth Intl. Conference on Document Analysis and Recognition*. 1999. Bangelore, India.
8. van Rijsbergen, C.J., *Information Retrieval*. 2nd ed. 1979, London: Butterworths.
9. Chu, H. and M. Rosenthal. *Search engines for the world wide web: a comparative study and evaluation methodology.* in *Annual Conference for the American Society for Information Science*. 1996.
10. Downie, J.S. *Panel on Music Information Retrieval Evaluation Frameworks.* in *3rd International Conference on Music Information Retrieval (ISMIR)*. 2002. Paris, France.
11. Reiss, J.D. and M.D. Sandler. *Benchmarking Music Information Retrieval Systems.* in *JCDL Workshop on the Creation of Standardized Test Collections, Tasks, and Metrics for Music Information Retreival (MIR) and Music Digital Library (MDL) Evaluation*. 2002. Portland, Oregon.
12. Reiss, J.D., J.-J. Aucouturier, and M.B. Sandler. *Efficient Multidimensional Searching Routines for Music Information Retrieval*. in *2nd Annual International Symposium on Music Information Retrieval*. 2001. Bloomington, Indiana, USA.
13. Schreiber, T., *Efficient neighbor searching in nonlinear time series analysis.* Int. J. of Bifurcation and Chaos, 1995. **5**(2): p. 349-358.
14. Liu, C.C., J.-L. Hsu, and A.L.P. Chen, *Efficient Near Neighbor Searching Using Multi-Indexes for Content-Based Multimedia Data Retrieval.* Multimedia Tools and Applications, 2001. **13**(3): p. 235-254.
15. Bentley, J.L. *K-d trees for semidynamic point sets.* in *Sixth Annual ACM Symposium on Computational Geometry*. 1990. San Francisco.
16. Moore, A.W., *Efficient Memory-based Learning for Robot Control*, in *Computer Laboratory*. 1991, University of Cambridge: Cambridge.
17. Rossignol, S., et al., *Automatic characterisation of musical signals: feature extraction and temporal segmentation*. Journal of new music research, 1999. **28**(4): p. 281 - 295.
18. Yianilos, P.N. *Data Structures and Algorithms for Nearest Neighbor Search in General Metric Spaces*. in *Fourth Annual ACM/SIGACT-SIAM Symposium on Discrete Algorithms*. 1993. Austin, Texas.
19. Beckmann, N., et al. *The R*-tree: an efficient and robust access method for points and rectangles*. in *ACM SIGMOD International Conference on Management of Data*. 1990. Atlantic City, New Jersey, United States: ACM Press.
20. Guttman, A. *R-trees: A dynamic index structure for spatial searching*. in *ACM SIGMOD Int. Conf. on Management of Data*. 1984. Boston, Massachusetts.
21. Lemström, K. and S. Perttu. *SEMEX - An Efficient Music Retrieval Prototype*. in *First International Symposium on Music Information Retrieval (ISMIR)*. 2000. Plymouth, Massachusetts.
22. Lemström, K. and J. Tarhio. *Searching monophonic patterns within polyphonic sources*. in *The RIAO Conference*. 2000. Paris, France.
23. Faloutsos, C., M. Ranganathan, and Y. Manolopoulos. *Fast Subsequence Matching in Time-Series Databases*. in *ACM SIGMOD Conference on Management of Data*. 1994. Mineapolis, MN, USA.
24. Wang, H. and C.-S. Perng. *The S^2-Tree: An Index Structure for Subsequence Matching of Spatial Objects*. in *5th Pacific-Asic Conference on Knowledge Discovery and Data Mining (PAKDD)*. 2001. Hong Kong: Springer-Verlag Berlin Heidelberg 2001.
25. Hearst, M., *User Interfaces and Visualization*, in *Modern Information Retrieval.* 1999, Addison-Wesley Longman Publishing Company.