Music-Notation Searching and Digital Libraries

Donald Byrd
CIIR, Department of Computer Science
University of Massachusetts
dbyrd@cs.umass.edu

ABSTRACT

Almost all work on music information retrieval to date has concentrated on music in the audio and event (normally MIDI) domains. However, music in the form of notation, especially Conventional Music Notation (CMN), is of much interest to musically-trained persons, both amateurs and professionals, and searching CMN has great value for digital music libraries. One obvious reason little has been done on music retrieval in CMN form is the overwhelming complexity of CMN, which requires a very substantial investment in programming before one can even begin studying music IR. This paper reports on work adding music-retrieval capabilities to Nightingale®, an existing professional-level music-notation editor.

1. INTRODUCTION

In recent years, interest in music information retrieval has been growing at a tremendous pace. The first meeting devoted exclusively to music IR was held late last year [14]; Byrd and Crawford [6] list much more evidence of the growth of interest in terms of grants and papers. There are three basic representations of music: audio, events (normally MIDI), and notations of various sorts. Almost all work on music IR to date has concentrated on the first two domains. However, music in the form of notation, especially the Conventional Music Notation (CMN) of Western society, is of much interest to musically-trained persons, both amateurs and professionals, so searching CMN has great importance for digital music libraries. Of the total music holdings of the Library of Congress, estimated at well over 10,000,000 items, there are believed to be over 6,000,000 pieces of sheet music and tens of thousands, perhaps hundreds of thousands, of scores of operas and other major works [15]. The sheet music and scores are all, of course, in some form of music notation, and the vast majority are undoubtedly in CMN. It is obvious that mechanical assistance could be invaluable in searching a collection of such magnitude.

It seems clear that a major reason little has been done on music retrieval in CMN form is the overwhelming complexity of CMN, which requires a very substantial investment in programming before one can even begin studying music IR. As evidence of its complexity, the source code for Nightingale®, an existing professional-level music-notation editor, amounts to some 160,000 lines of C. We will have more to say about the complexity of CMN.

Another likely reason for the dearth of music-retrieval work on CMN is a lack of collections with which to experiment. The practical availability of what CMN exists in machine-readable form is seriously hampered by the fact that, nothwithstanding several attempts at a standardized format for CMN representations of music [7], no effective standard exists. But the lack of CMN collections is likely to change soon, especially in view of work like the Levy sheet-music project at Johns Hopkins University [8], which is applying Optical Music Recognition on a large scale to create a CMN collection.

This paper reports on work adding music-retrieval capabilities to Nightingale.

2. BACKGROUND

2.1. Basic Representations of Music and Audio

The material in this section is an abridgement of the section of the same title in [6].

There are three basic representations of music and audio: the well-known *audio* and *music notation* at the extremes of minimum and maximum structure respectively, and the less-well-known *time-stamped events* form in the middle. Numerous variations exist on each representation. All three

This material is based on work supported in part by the Digital Libraries Initiative, Phase 2, under NSF grant IIS-9905842, by the U.K. Joint Information Systems Committee under project code JCDE/NSFKCL, and by the National Science Foundation, Library of Congress and Department of Commerce under cooperative agreement number EEC-9209623.

Any opinions, findings and conclusions or recommendations expressed in this material are the author's and do not necessarily reflect those of the sponsors.

JCDL'01, June 24-28, 2001, Roanoke, Virginia, USA.

are shown schematically in Figure 1, and described in Figure 2.

The “Average relative storage” figures in the table are for uncompressed material and are our own estimates. A great deal of variation is possible based on type of material, mono vs. stereo, etc., and—for audio—especially with such sophisticated forms as MP3, which compresses audio typically by a factor of 10 or so by removing perceptually unimportant features.

“Convert to left” and “Convert to right” refer to the difficulty of converting fully automatically to the form in the column to left or right. Reducing structure with reasonable quality (convert to left) is much easier than enhancing it (convert to right).

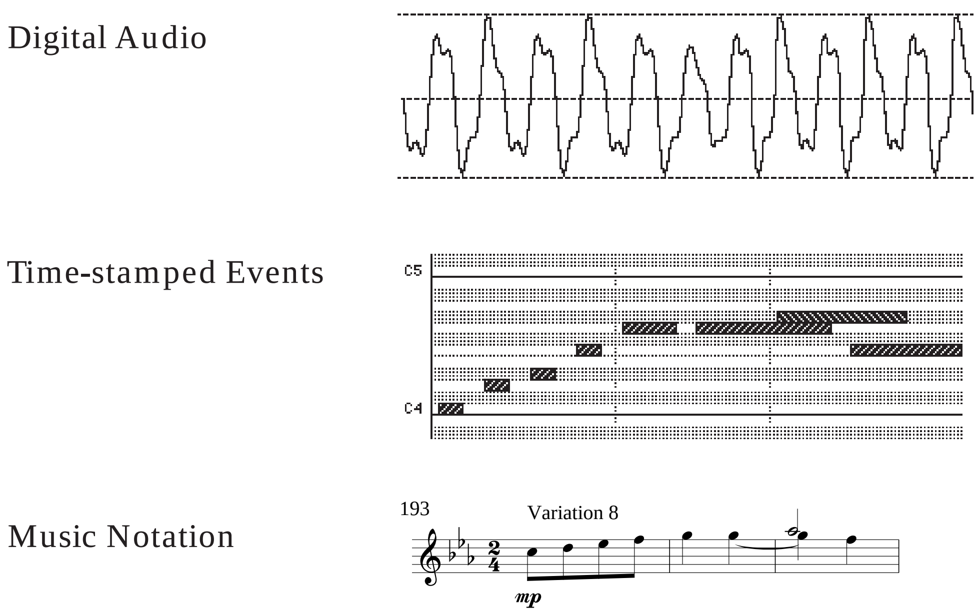

Fig. 1. Basic representations of music

Representation	*Audio*	*Time-stamped Events*	*Music Notation*
Common examples	CD, MP3 file	Standard MIDI File	sheet music
Unit	sample	event	note, clef, lyric, etc.
Explicit structure	**none**	**little (partial voicing information)**	**much (complete voicing information)**
Avg. rel. storage	2000	1	10
Convert to left	-	easy	OK job: easy
Convert to right	1 note/time: pretty easy; 2 notes/time: hard; other: very hard	OK job: fairly hard	-
Ideal for	music bird/animal sounds sound effects speech	music	music

Fig. 2. Basic representations of music

2.1.1. *Music Notation*

There is little doubt that CMN is among both the most elaborate and the most successful graphic communication schemes ever invented. Its complexity places great demands on developers of music-notation software: we have already mentioned the amount of code Nightingale requires. For details of CMN, see standard texts such as those by Read [21] and Ross [22]. For a discussion of its complexity and the implications for software, see [4], especially Chapter 2, and [5].

The success of CMN is obvious from the facts that it has survived with relatively minor changes for over 300 years (see for example [20], pp. 15 ff.), and that it has withstood numerous attempts at major overhaul or complete replacement (see "Notation", Sec. III.4.v, in [23]). Nonetheless, there are other established notations for music, for example tablature (mostly for guitar, lute, and similar instruments: see [20], pp. 143–171), Braille (for blind musicians), and the notations of such other cultures as China, India, Indonesia, and Japan; these systems are beyond the scope of this paper.

2.1.2. *Multiple Representations in Music-IR Systems*

It is important to realize that, in a music-IR system, the internal representation and the external representation—the form used in all aspects of the user interface—may be different; in fact, a system might use a different form in the query and document-display interfaces. In particular, a system might deal with event-level databases, yet accept queries and/or display results in notation form. In an extreme case, it might accept queries in notation form, search an audio database, and display results in a graphic display of events in retrieved audio documents.

2.2. OMRAS and This Work

This work is part of the OMRAS (Online Music Recognition and Searching) project [19]. Among the major goals of OMRAS is to handle music in all three basic representations discussed above with as much flexibility as possible. We are working on searching databases of polyphonic music in all three basic representations, with a full GUI for complex music notation. But beyond this, we are attempting to maximize flexibility with a modular (plug-in) architecture, and exploiting that flexibility by developing and testing two systems with different representations, search methods, and user interfaces (my own NightingaleSearch, and Matthew Dovey's Java Musical Search (JMS) [11]. We feel that the three basic representations can be usefully combined in several ways. Most relevant here is that even when the database is in audio or MIDI form, for many people, CMN will still be useful for formulating queries and displaying retrieved documents. (Admittedly, this is not always practical. As we have said, converting MIDI to CMN for display purposes is not easy, and converting audio to CMN for display is a great deal harder.)

Other threads of the OMRAS project that should eventually interact with CMN-based retrieval work are research on recognition of music from polyphonic audio [3] and research on efficient algorithms for searching music [10].

2.3. Related Work

The research most closely related to this is probably Donncha O'Maidin's C.P.N.View [17, 18]. However, O'Maidin has concentrated on folk music, and his system appears to handle only simple monophonic music. McNab's MR system—part of the MELDEX project—maintains a database in notation form, and it can display both queries and melodies it retrieves in CMN [16, 2]. But again it can handle only simple monophonic music, in this case without tuplets, beams, etc. Furthermore, queries must be entered in audio form: there is no CMN entry or editing.

The well-known commercial music editor Finale has for years had a command for searching music in CMN form by content, but it can search only within a single score at a time [9]. Perhaps more important, Finale limits itself to what might be called "document-editor" style searching, i.e., finding the next match for Boolean criteria. This is as opposed to the "IR" style searching for all matches in a document or database that makes possible best-match IR and ranking.

In fact, work on music retrieval in CMN form is conspicuous thus far by its scarcity. The obvious reason is the huge investment in programming complex CMN demands before one can even begin studying music IR.

So-called "piano roll" notation is the graphic equivalent of music in the event representation. For complex music, piano roll is a great deal less demanding than CMN, and it can convey much of the same information; but it has not been used in music IR much, either. One system that does use piano roll, albeit in a simplified form indicating note onsets but not durations, is Dovey's, in his testbed framework Java Music Search (JMS) [11]. Dovey not only displays both queries and retrieved music in this form, he also uses it as an abstract model of music.

2.4. Music Information Needs and the Audience for Searching CMN

It seems obvious that—in the face of MIDI and, especially, audio as alternatives—CMN as a basis for a music-retrieval system will be of interest only to those with some knowledge of CMN. On the other hand, for Western music of the last few centuries, at least, CMN is arguably the best graphic representation ever developed: it has value purely as a user-interface device.

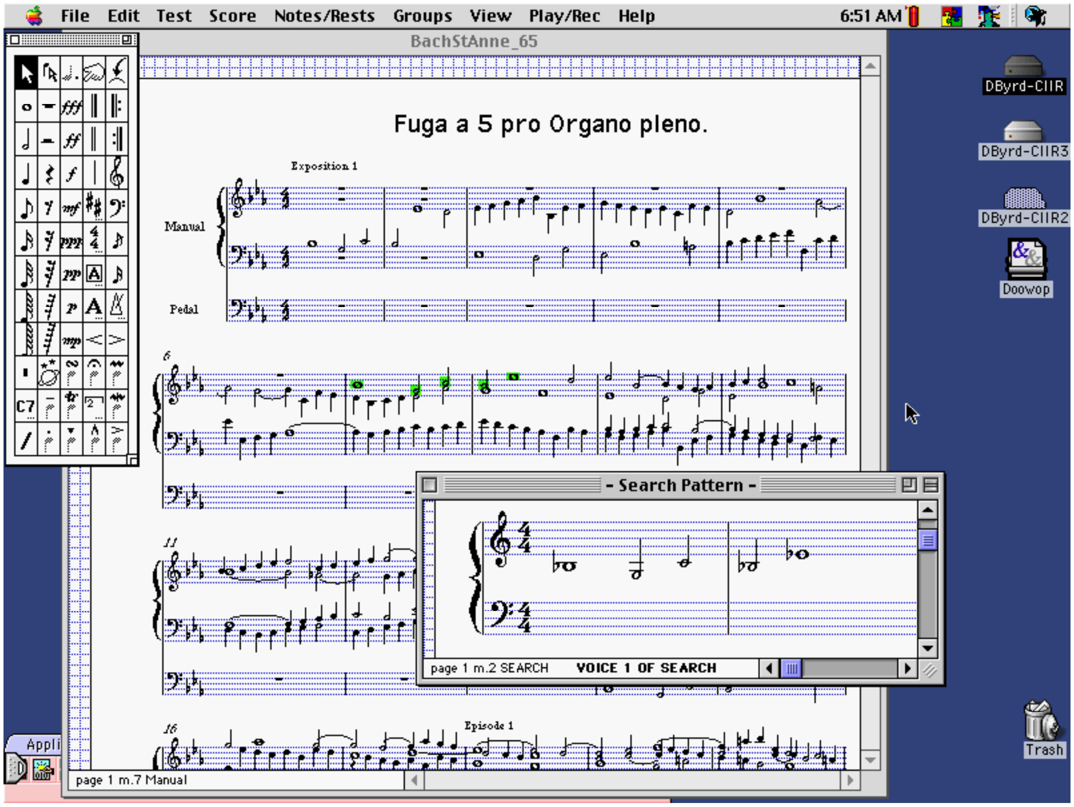

Fig. 3. Bach: "St. Anne's" Fugue, with Search Pattern

3. NIGHTINGALESEARCH

Nightingale® is a professional-level music-notation editor for the Macintosh computer, written in the C language; it has been marketed commercially for a number of years [1]. Since I led the team that developed Nightingale, I not only had access to the source code, I knew it well. I decided to use it as a platform for studying CMN-based music IR by adding several music-searching features and commands: the resulting program is "NightingaleSearch".

3.1. Overview

NightingaleSearch inherits all the normal functionality of Nightingale. It can display and edit any number of *scores*—CMN documents—at the same time, and it supports several ways of creating music, including recording from a MIDI device (usually a synthesizer keyboard), importing standard MIDI files, pasting from other scores, etc. The searching commands use the contents of a special score, the "Search Pattern", as the query. In nearly all respects, this is an ordinary Nightingale score, and music can be entered into it with any of Nightingale's facilities. See Figure 3.

Menu commands to "Search for Notes/Rests" and "Search in Files" bring up the dialog in Figure 4. NightingaleSearch is a research prototype, and I show the dialog only to make clear what the program can do: there are far too many options for mortal users. To sum it up, matching can be based on pitch, duration, or both. In IR terms, matching is Boolean: there are no approximate matches, except for those allowed by Tolerance (for pitch) and preserve contour (for duration) as described below. The main options are:

- **Match pitch (via MIDI note number)**: if not checked, matching ignores the pitches of the notes. *Relative* matches any transposition of the entire pattern; *absolute* matches only the exact original pitches. Pitch options include:
    - Tolerance: each interval can be off from the corresponding interval in the pattern by the given number of semitones. However, for "relative" matches, if "always preserve contour" is checked, the match will still fail unless the upward, repeat, or downward motion of each interval in the pattern is preserved. This is very useful to avoid "false positives": without it, for example, a tolerance of 2 would allow an upward chromatic scale to match a downward one or a series of repeated notes.
- **Match duration (notated, ignoring tuplets)**: if not checked, matching ignores the durations of the notes and—if rests are included—of the rests. *Relative* matches the original series of durations multiplied by

any factor: in musical terms, it recognizes augmentation and diminution. *Absolute* matches only the exact original series of durations. Duration options include:

- Preserve contour: this is analogous to the "always preserve contour" option for pitch in that it distinguishes just three relationships (in this case, longer, shorter, and the same), though it differs by being an alternative to relative or absolute rather than modifying relative.

- **In chords, consider**: all notes, outer notes only, or top note only. Notice that a chord in Nightingale is entirely within a voice, so these options do not apply, say, to a brass quintet where each instrument plays a single note: they are mostly for keyboard music. In any case, "all notes" will rarely be useful, since inner notes of chords nearly always serve just to enrich the harmony or texture.

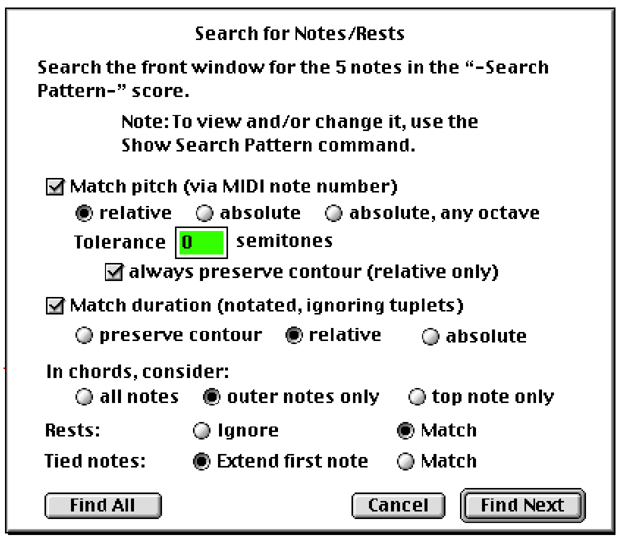

Fig. 4. Search Dialog

Search for Notes/Rests just searches the score in the frontmost window. Search in Files is more interesting. It exists in a version that searches all Nightingale scores in a given folder, and a version that searches a "database". As of this writing, the database is simply a file that describes in order of occurrence all the notes in any number of Nightingale scores, with information identifying the original scores. Thus, it does not provide a way "to avoid the efficiency disaster of sequential searching". [6] Text IR gets around this problem by indexing, which can improve performance with a large database by thousands of times; research on indexing polyphonic music is underway or planned by several groups, including OMRAS.

3.1.1. Retrieval Levels and the Result List

NightingaleSearch does passage-level retrieval, i.e., it looks for and reports individual occurrences of matches for the search pattern. In contrast, most IR systems, for music as well as text, retrieve entire documents that match the pattern in one or more places. It could be argued that the "average" music document is much longer and more complex than the "average" text document, and therefore retrieval of passages is much more important with music. This is a strong argument, though of course it depends on the document collection: by any obvious measure, the average article in *The New Yorker* is longer than the average folksong.

Currently, the result list is displayed in a scrolling-text window; there is no link to let the user choose an entry in the list and view that "match" in CMN. MELDEX [2] lets the user listen to any entry in its result list as well as view it, and both options would be very helpful for NightingaleSearch.

3.2. NightingaleSearch in Action

Notation representations of music—CMN or other—are distinguished from audio and event representations mostly by the amount of explicit structure they contain. In particular, with minor exceptions, music in CMN contains complete voicing information, i.e., the voice membership of every note is evident from the notation. For example, the opening of Bach's "St. Anne's" Fugue is shown in Figure 3: the three staves contain five voices, as suggested by the stems going up and down for notes on the upper two staves. The first five notes of the piece are enough for a human musician to identify all 20 or so clearcut occurrences of the main subject (essentially, the theme), but searching for exact (except for transposition) matches of them finds only 5, all valid. This is 100% precision but only 25% recall. One problem is that some instances are so-called "tonal answers", resulting in pitch intervals slightly different from the original. For example, the second occurrence of the subject, starting in m. 3, begins by going down 1 semitone rather than the original version's 3. Setting the tolerance in the search dialog to 2 results in finding 8 matches: again all are valid, but 12 valid "hits" were still not found, for a precision of 100% and recall of 40%. The result list appears in Figure 5. Notice that, for each match, NightingaleSearch displays a label for the section of the piece (the passage) as well as the measure number, plus the voice number and "instrument" (actually, "Manual" and "Pedal" are both parts of the single instrument this piece was written for, the organ). This much information is very rarely available in event representations, and never in audio.

Time 0.13 sec. 8 matches (in order of error):
1: BachStAnne_65: m.1 (Exposition 1), voice 3 of Manual, err=p0 (100%)
2: BachStAnne_65: m.7 (Exposition 1), voice 1 of Manual, err=p0 (100%)
3: BachStAnne_65: m.14 (Exposition 1), voice 1 of Pedal, err=p0 (100%)
4: BachStAnne_65: m.22 (Episode 1), voice 2 of Manual, err=p0 (100%)
5: BachStAnne_65: m.31 (Episode 1), voice 1 of Pedal, err=p0 (100%)
6: BachStAnne_65: m.26 (Episode 1), voice 1 of Manual, err=p2 (85%)
7: BachStAnne_65: m.3 (Exposition 1), voice 2 of Manual, err=p6 (54%)
8: BachStAnne_65: m.9 (Exposition 1), voice 4 of Manual, err=p6 (54%)

Figure 5. Result list for search of the "St. Anne's" Fugue

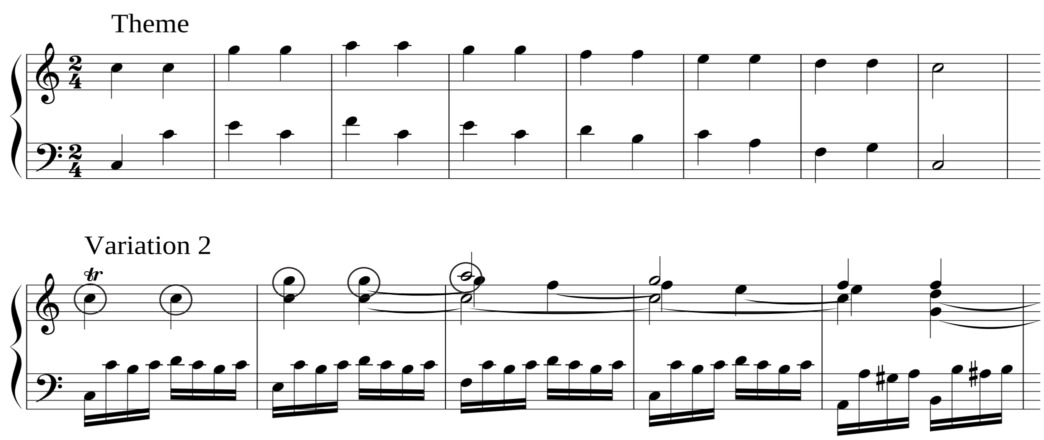

Fig. 6a (above) and b (below) (Mozart)

Using more of the fugue subject as the query naturally tends to increase precision at the expense of recall. However, with the first seven notes of the piece as query, tolerance of 2, and ignoring duration, it does well on both metrics: it finds 22 matches, of which 4 are false, for a precision of 82% and recall of 90%.

For another example, consider a user looking in a digital music library for the old children's song that is called in English-speaking countries by several names, but best known as "Twinkle, Twinkle, Little Star". This melody has been used in many ways, including music by (among others) Mozart, Dohnanyi, and the violin pedagogue Shinichi Suzuki. Mozart used it in his Variations for piano, K. 265, on "Ah, vous dirais-je, Maman"; the melody is shown in his version in Figure 6a. One difficulty this piece demonstrates is the effects of complete voicing on music IR. In Variations 2 (Figure 6b), 4, and 9, the melody starts in one voice, then, after four notes—not enough for a reliable match—moves to another. Of course, it is easy simply to ignore voice information, but doing so is likely to have catastrophic effects on precision [6].

In fact, this piece of Mozart's demonstrates several difficult problems for music IR. Some of the other variations employ tricks like distorting the melody or adding ornamental notes to it, but others discard the melody completely while retaining the harmony and bass line! But none of these subtleties really matters to our hypothetical digital-music-library user, who presumably simply needed their attention drawn to the Mozart piece: in other words, document-level retrieval is adequate in this case. Searching for the first four notes of the Twinkle theme in a very small database finds the matches shown in Figure 7.

Time 1.27 sec. 13 matches (in order found):
1: BaaBaaBlackSheep: m.1, voice 1 of Unnamed
2: BaaBaaBlackSheep: m.9, voice 1 of Unnamed
3: Mozart-TwinkleVar_10: m.1 (Theme), voice 1 of Piano
4: Mozart-TwinkleVar_10: m.84 (Variation 9), voice 2 of Piano
5: Suzuki-TwinkleVar: m.16 (Variation D), voice 1 of Violin
6: Suzuki-TwinkleVar: m.21 (Theme), voice 1 of Violin
7: Suzuki-TwinkleVar: m.29 (Theme), voice 1 of Violin
8: Twinkle-Hirsch2ndGraderVer: m.1, voice 1 of Unnamed
9: Twinkle-Hirsch2ndGraderVer: m.9, voice 1 of Unnamed
10: TwinkleHARMONETVar: m.1, voice 1 of Original
11: TwinkleHARMONETVar: m.9, voice 1 of Original
12: TwinkleMelody: m.1, voice 1 of Unnamed
13: TwinkleMelody: m.9, voice 1 of Unnamed

Figure 7. Result list for search for the "Twinkle" theme

3.3. Intuition vs. Evaluation in Music IR

No formal evaluation has yet been done of NightingaleSearch. In fact, a great deal of work on music IR to date has been speculative, and what evaluation of systems has been done has generally not been at all rigorous. It is tempting to criticize researchers for their unscientific work, but, in the words of Byrd and Crawford [6] (citations omitted):

> To put things in perspective, music IR is still a very immature field... For example, to our knowledge, no survey of user needs has ever been done (the results of the European Union's HARMONICA project are of some interest, but they focused on general needs of music libraries). At least as serious, the single existing set of relevance judgements we know of is extremely limited; this means that evaluating music-IR systems according to the Cranfield model that is standard in the text-IR world…is impossible, and no one has even proposed a realistic alternative to the Cranfield approach for music. Finally, for efficiency reasons, some kind of indexing is as vital for music as it is for text; but the techniques required are quite different, and the first published research on indexing music dates back no further than five years. Overall, it is safe to say that music IR is decades behind text IR.

I would argue that the state of the art of music-IR evaluation is so primitive, there is little point in trying to evaluate music-IR systems and techniques rigorously. Instead, the field is best served by music-IR system developers relying on intuition and informal evaluation, while other researchers develop tools to make meaningful evaluation possible.

4. CONCLUSIONS

Other than Finale—which is limited to finding a single match at a time in a single file—NightingaleSearch is the only program I know of that allows searching complex music with a query in any type of music notation, and the only program that displays the results of such a search in notation form. NightingaleSearch has many shortcomings. Not the least is that any music to be searched must first be in a format it can use, but we are working on connectivity with other programs, for example, via a utility that converts music in the well-known Humdrum kern format [13]. Also, any evaluation of NightingaleSearch, even the most basic, remains to be done. In any case, there would not be much point to evaluating it with the primitive tools available now. But informal use to date strongly supports intuitions of the value of notation-based music retrieval. In the not-too-distant future, the ability to search music notation will surely be part of every digital music library that contains notation.

5. ACKNOWLEDGEMENTS

In addition to their other contributions to the OMRAS project, Tim Crawford, Matthew Dovey, and Jeremy Pickens made many helpful comments on this paper. Thanks also to Alan Belkin for his comments.

6. REFERENCES

[1] AMNS (2000). Nightingale. http://www.ngale.com .

[2] Bainbridge, D. (1998). MELDEX: A Web-based Melodic Index Service. In Hewlett & Selfridge-Field, *op. cit.*

[3] Bello, J.P., Monti, G., and Sandler, M. (2000). Techniques for Automatic Music Transcription. Read at the First International Symposium on Music Information Retrieval; available at http://ciir.cs.umass.edu/music2000 .

[4] Byrd, D. (1984). *Music Notation by Computer.* PhD dissertation, Indiana University (available from UMI, Ann Arbor, Michigan, U.S.A.).

[5] Byrd, D. (1994). Music Notation Software and Intelligence. *Computer Music Journal* 18(1), pp. 17–20.

[6] Byrd, D. and Crawford, T. (2001). Problems of Music Information Retrieval in the Real World. To appear in *Information Processing and Management.*

[7] Castan, G. (2001). Musical notation codes. http://www.s-line.de/homepages/gerd_castan/compmus/notationformats_e.html .

[8] Choudhury, G.S., Droetboom, M., DiLauro, T., Fujinaga, I., and Harrington, B. (2000). Optical Music Recognition System within a Large-Scale Digitization Project. Read at the First International Symposium on Music Information Retrieval; available at http://ciir.cs.umass.edu/music2000 .

[9] Coda (2000). *About Finale for Macintosh* (Coda Music Technology).

[10] Crawford, T., Iliopoulos, C.S., and Raman, R. (1998). String-Matching Techniques for Musical Similarity and Melodic Recognition. In Hewlett and Selfridge-Field, *op. cit.*

[11] Dovey, M. (1999). A matrix based algorithm for locating polyphonic phrases within a polyphonic musical piece. In *Proceedings of AISB '99 Symposium on Artificial Intelligence and Musical Creativity*. Edinburgh, Scotland: Society for the Study of Artificial Intelligence and Simulation of Behaviour.

[12] Hewlett, W., and Selfridge-Field, E., eds. (1998). *Melodic Similarity: Concepts, Procedures, and Applications (Computing in Musicology 11)* (MIT Press).

[13] Huron, D. (1998). Humdrum and Kern: Selective Feature Encoding. In Hewlett & Selfridge-Field, *op. cit.*

[14] ISMIR (2000). International Symposium on Music Information Retrieval. http://ciir.cs.umass.edu/music2000 .

[15] LaVine, K. (2000). Personal communication, May 2, 2000.

[16] McNab, R., Smith, S., Bainbridge, D., and Witten, I. (1997). The New Zealand Digital Library MELody InDEX. DLib Magazine, May 1997; at http://www.dlib.org .

[17] O'Maidin, D. (1995). "A Programmer's Environment for Music Analysis". Technical Report UL-CSIS-95-1, Department of Computer Science, University of Limerick, Ireland.

[18] O'Maidin, D. (1998). "A Geometrical Algorithm for Melodic Difference." In Hewlett & Selfridge-Field, *op. cit.*

[19] OMRAS (2000). Online Music Recognition and Searching. http://www.omras.org .

[20] Rastall, R. (1982). *The Notation of Western Music* (St. Martin's Press).

[21] Read, G. (1969). *Music Notation,* 2nd ed. (Crescendo).

[22] Ross, T. (1970). *The Art of Music Engraving and Processing* (Hansen).

[23] Sadie, S., ed. (1980). The New Grove Dictionary of Music and Musicians (Macmillan).