Journal of New Music Research

ISSN: 0929-8215 (Print) 1744-5027 (Online) Journal homepage: http://www.tandfonline.com/loi/nnmr20

Towards a Standard Testbed for Optical Music Recognition: Definitions, Metrics, and Page Images

Donald Byrd & Jakob Grue Simonsen

To cite this article: Donald Byrd & Jakob Grue Simonsen (2015) Towards a Standard Testbed for Optical Music Recognition: Definitions, Metrics, and Page Images, Journal of New Music Research, 44:3, 169-195, DOI: 10.1080/09298215.2015.1045424

To link to this article: http://dx.doi.org/10.1080/09298215.2015.1045424

Journal of New Music Research, 2015
Vol. 44, No. 3, 169–195, http://dx.doi.org/10.1080/09298215.2015.1045424

Routledge
Taylor & Francis Group

Towards a Standard Testbed for Optical Music Recognition: Definitions, Metrics, and Page Images

Donald Byrd[1] and Jakob Grue Simonsen[2]

[1]*Indiana University, USA;* [2]*Department of Computer Science, University of Copenhagen (DIKU), Denmark*

(Received 5 July 2014; accepted 7 April 2015)

Abstract

We posit that progress in Optical Music Recognition (OMR) has been held up for years by the absence of anything resembling the standard testbeds in use in other fields that face difficult evaluation problems. One example of such a field is text information retrieval (IR), where the Text Retrieval Conference (TREC) has annually-renewed IR tasks with accompanying data sets. In music informatics, the Music Information Retrieval Exchange (MIREX), with its annual tests and meetings held during the ISMIR conference, is a close analog to TREC; but MIREX has never had an OMR track or a collection of music such a track could employ. We describe why the absence of an OMR testbed is a problem and how this problem may be mitigated. To aid in the establishment of a standard testbed, we provide (1) a set of definitions for the complexity of music notation; (2) a set of performance metrics for OMR tools that gauge score complexity and graphical quality; and (3) a small corpus of music for use as a baseline for a proper OMR testbed.

Keywords: optical music recognition, empirical evaluation, notation, notation complexity

1. Introduction

There is now a large body of scientific work on Optical Music Recognition (OMR), and both commercial and free OMR systems have been available for many years. A question of interest to quite a few people is: what is the most *accurate* OMR system, at least for a specific collection of page images? Unfortunately, the answer is 'No one knows, and there's no practical way to find out'. The reason is that OMR evaluation is still very much a subject of contention (Rebelo et al., 2012), and we believe it will remain that way until a *standard testbed* exists for it.

Over 20 years have passed since the groundbreaking study of OMR systems at the Center for Computer-Assisted Research in the Humanities (henceforth 'CCARH'; Selfridge-Field et al., 1994), and it is not clear how much progress has been made on OMR systems since then. Anecdotal evidence suggests that substantial progress has been made, but, since evaluation of OMR systems is so unsettled, no one can be certain. At least one reason for this state of affairs is easy to find. It is well known in the document-recognition community that the difficulty of *evaluating* document-recognition systems (as opposed to the difficulty of *creating* them) varies enormously (Kanai et al., 1993; Nagy, 1995). The most familiar type of recognition system, Optical Character Recognition (OCR), is probably among the easiest to evaluate; but the system to recognize conventional Western music notation is undoubtedly among the hardest. (For discussion of the reasons, see Section 4 below.)

Droettboom and Fujinaga (2004) write that 'It could be argued that a complete and robust system to evaluate OMR output would be almost as complex and error-prone as an OMR system itself'. Similarly, Bainbridge and Bell (2001, p. 115) comment 'Perhaps the hardest task in OMR is comparing competing systems'. But other disciplines – among them, text IR, speech recognition, and even music IR (e.g. for audio chord estimation) – have faced very difficult evaluation problems and solved them. In every case we are aware of, the solution has been the same: the interested community has established standard testbeds, with well-thought-out test collections and evaluation metrics; run regular 'contests' (what are sometimes called 'campaigns') for researchers in the field; and held a conference after each 'contest' in which the participants report on and discuss what happened. Text IR may

Correspondence: Department of Computer Science, University of Copenhagen. Njalsgade 128–132, DK-2300 Copenhagen S, Denmark. E-mail: simonsen@diku.dk

This work was supported in part by the Andrew W. Mellon Foundation.

Table 1. Levels of description of CWMN.

Level	*Bellini et al. term*	*Examples*	*Musical construct ?*
pixel	–		no
low-level symbol	basic symbol	noteheads, flags, the letter ‘*p*’	no
high-level symbol	composite symbol	8th notes with flags, chords,dynamic marks ‘*p*’ and ‘*pp*’	yes
score	–	Schumann: *Fantasiestücke*, Henle ed.	yes

be the earliest example, with its TREC (Text REtrieval Conference) series (trec.nist.gov). One might guess that OCR would have been earlier, but, to our knowledge, there have never been regular OCR ‘contests’; this is perhaps unsurprising as OCR evaluation is fairly straightforward.

The U.S. National Institute of Standards and Technology established annual TREC conferences, with ‘tracks’ for specific text-IR tasks, over 20 years ago. In music informatics, MIREX and its annual tests, held in conjunction with the annual ISMIR (International Society for Music Information Retrieval) conferences, is a close analog to TREC; but MIREX has never had an OMR track, and it is doubtful that the ‘well-thought-out test collections and evaluation metrics’ materials such a track would need to exist.

The purpose of this paper is to consider why no OMR testbed exists, why that is a problem, and how the problem may be mitigated. As background, we discuss the challenges of OMR in some detail. To aid in the future establishment of a standard testbed, we provide (1) a set of definitions for the complexity of music notation; (2) a set of performance metrics for OMR tools that gauge score complexity and graphical quality; and (3) a small corpus of music for use as a baseline for a proper OMR testbed.

1.1 Terminology and scope

In common with the vast majority of work on OMR to date, we focus exclusively on Conventional Western Music Notation, which henceforth is abbreviated to *CWMN*. The phrases ‘Common Music Notation’ and ‘Traditional Music Notation’ are often used for the same thing.

Like some previous workers in OMR, we use the term *metrics* loosely, to mean a set of measurement guidelines that, when applied, yields a numeric evaluation of something.

2. Related work

2.1 Evaluation in text retrieval, music retrieval, and OMR

Cyril Cleverdon introduced the basic methodology for evaluating text information-retrieval systems as long ago as the mid-1960s, in the well-known Cranfield studies (Cleverdon, 1967). Cleverdon’s method has been adopted by TREC, and with minor changes by MIREX. It requires three components: a standardized *task*; a standardized *corpus of test documents*; and a standardized *evaluation method* based, naturally, on ground truth. For information retrieval, the task is usually finding documents that satisfy predefined information needs (informally, queries). Given the ground truth of what documents are relevant to each query, the evaluation method is generally measuring recall for several values of precision or vice versa. For document recognition, the appropriate task is, of course, the accuracy of the symbolic representation generated for each document. Evaluating OMR accuracy is vastly more difficult than computing recall and precision; it is discussed at length in the remainder of the current paper.

2.2 Work on evaluation in OMR

To our knowledge, the first serious attempt to evaluate OMR systems was at the CCARH in the early and mid-1990s; see the substantial report by Selfridge-Field et al. (1994). Selfridge-Field and her colleagues performed what appears to be the first categorization of problems for OMR tools, grouping problems into three categories: visual surface problems, object recognition problems, and problems of music representation.

> The first category, *visual surface problems*, roughly pertains to problems related to image quality, dirt, and other visual damage. The category of *object recognition problems* corresponds to ‘systemic’ problems due to either typesetting (e.g. different fonts being used, or superimposition of symbols) or to factors inherent in the notation system (e.g. the presence of grace notes).
> The *problems of music representation* category considers the various anomalies of common Western music notation (e.g. common stems for notes of different duration), and superfluous symbols not intended for recognition (e.g. guitar chords set below or above the staff).

All of these problems remain relevant and the grouping into three categories remains useful. However, for the purpose of quantitatively comparing the output of OMR systems, this taxonomy is not adequate. Selfridge-Field et al. made no attempt to stratify problems into various grades (as will be made clear later in the present paper, this issue is crucial for proper comparisons), and the list of problems within each category is not quite exhaustive (see the Sections 3 and 4 below). Much of the work reported below may be seen as attempts to further develop and qualify their categories; in particular, the grading of images and music we devise for OMR systems can be viewed as attempts to quantify the issues belonging to the categories of ‘visual surface problems’, and to equip the categories ‘object recognition’ and ‘music representation’ with numeric levels of difficulty.

Selfridge-Field et al. had previously sent some 36 OMR system developers two sample music images – essentially, a tiny standard test collection – with a questionnaire asking how difficult various features of the notation were and how well their systems would handle them. They received responses from seven developers, and all are summarized in their report. In addition, they raised the issue of how practical OMR really is, using Haydn's Symphony No. 1 (in a 1907 Breitkopf and Härtel edition) to compare the amount of time needed for hand encoding to the time needed for OMR followed by hand corrections to its inevitable mistakes.

The case for a standard testbed for OMR has been made before (Byrd et al., 2010; Jones et al, 2008). Byrd et al. (2010, Section 4.2) spells out the problems with the current state of OMR evaluation:

> It is clear that the state of [OMR evaluation] is dismal, and one reason, we have argued, is that it is inherently very difficult. An appropriate evaluation metric is, to put it mildly, not straightforward... As a first step, the existence of a standard testbed – i.e., some reasonable (even if very imperfect) evaluation metrics, and a set of carefully selected music page images to apply them to – would at least allow some meaningful comparisons of OMR programs. But, to our knowledge, there are absolutely no standards of any kind in general use; that is the second reason current OMR evaluation is so unsatisfactory... If program A is tested with one collection of music and program B is tested with a different collection, and there is no basis for saying which music is more challenging, how can the results possibly be compared? What if one system is evaluated at a low level and the other at a high level? That alone might lead to a very different error rate. As things stand, there's no way to know if any OMR system…really is more accurate than any other for any given collection of music, much less for music in general.

(We define the terms *low level* and *high level* in Section 5.1.1 below.) In addition to the 'comparing apples and oranges' problems mentioned in the above quote, the vast majority of OMR studies we have seen say little about how they chose the pages in their test collections; the collections appear to be what statisticians would call 'convenience samples', chosen primarily because they were easily available. Exceptions include the 'SFStudy' collection of 24 pages, so called because the pages were first used for a 'Scoping and Feasibility Study' for the OMR component of the MeTAMuSE project (Byrd & Schindele 2006, 2007), and the 7-page collection of Bellini et al. (2003). The SFStudy collection has since been used by Bugge et al. (2011) in their own OMR work. We discuss this collection in detail in Section 8 below.

Many of the difficulties due to varying notational complexity and high- or low-level error counting have been described in prior literature, including Ng and Jones (2003), Droettboom and Fujinaga (2004), Bellini et al. (2007), Jones et al. (2008), and Byrd et al. (2010). However, none of these papers contain a set of detailed definitions for score complexity and image quality (with the ensuing implications for a testbed), and none describes an *easily adaptable* system for error counting.

Droettboom and Fujinaga present an approach to low-level OMR evaluation, but they leave the exact choice of metric to the reader, and they do not provide a corpus. Ng and Jones argue that, when comparing OMR systems, one should weight error counts to reflect the edit cost of correcting errors; this interesting idea goes back at least as far as Bainbridge and Bell (2001), and we discuss it in Section 5.2 below.

The important paper by Bellini et al. (2007) suggests an approach to OMR evaluation relying on similar components to ours. They advocate employing weights to assign relative importance to different symbols, and they give a specific weighting determined empirically, by experts' responses to a questionnaire. (Both the score-page images and the questionnaire are available online; see Bellini & Nesi, 2003, 2014.) No inter-rater agreement scores are provided, and they implicitly assume that the weight of each symbol is independent of its context in the score, which may not reflect the effort needed to correct the error. A more serious caveat is that users interested in different types of music may well assign weights to symbols very differently from these experts. Nonetheless, the basic idea is original and significant. Bellini et al. use these ideas to test three OMR programs using a very small corpus of seven score-page images. They do not consider score image quality: all seven images are of high quality. Jones et al. (2008) describe a multi-step procedure for evaluating OMR systems, and they use the corpus and weights of Bellini et al. to compare systems.

Knopke and Byrd (2007) and Szwoch (2008) advocate automating evaluation of OMR systems by comparing output MusicXML to ground-truth MusicXML and basing error counts on the discrepancies, thus greatly reducing the need for manual error counting. However, neither provides a test set or a discussion of music complexity or image quality. In addition, major obstacles to automating evaluation remain.

2.2.1 *Automated evaluation*

Thus far, every evaluation of OMR system accuracy we know of has been done entirely by hand, a process that is itself error-prone as well as time consuming – and, of course, the larger the test collection, the more time consuming it is. Both TREC and MIREX obtain ground truth manually, then automatically tally discrepancies between each system's results and ground truth. The idea of basing error counts on automatic tallies of discrepancies between MusicXML files produced by OMR programs is very appealing, but it requires overcoming several obstacles.

Until recently, automating evaluation for OMR systems was completely impractical because the only common formats for the score-level representations of the music produced by systems were the formats of such score editors as Finale and Sibelius, formats that were not designed for use by other programs. The advent of standard symbolic representations of CWMN changed things; most major OMR systems now export MusicXML. But, while the availability of output in a common format is a significant step towards automatic evaluation, it would still be extremely difficult. One reason is that there are multiple correct symbolic forms for the same notation. Perhaps more surprising is the experience of Knopke and Byrd (2007) that the systems they tested repeatedly mistook note stems for barlines and vice versa, thereby requiring global alignment of each system's output to ground truth: that is, the 'obvious' simplification of aligning ground truth and the OMR system output a bar at a time is very unreliable. For the near future, it appears that we are stuck with manual error counting.

2.3 Our work in perspective

While we build on much of the above-described work, especially that of Selfridge-Field et al., and of Bellini et al. and Jones et al., we present several things that apparently have not appeared in the literature of OMR evaluation before: (a) An extensive description of the problems for OMR systems inherent in CWMN. (b) A more in-depth discussion concerning the difficulty of assigning error counts and error weights (and the ensuing issues of 'comparing-apples-and-oranges'); both Bellini et al. (2003) and Jones et al. (2008) discuss these problems, but only briefly. (c) Definitions of grades for both image quality and notation complexity. (d) A sample corpus for OMR evaluation based on these ideas. At 34 pages, our sample corpus is not large, but it is much larger than those used in nearly all previous OMR studies we know of.

3. OMR and OMR evaluation

OMR is, roughly, the electronic conversion of scanned or photographed images of handwritten or printed sheet music into symbolic and therefore editable form. If one wishes to understand OMR evaluation, understanding how OMR works is very helpful. Several good accounts have been published: see for example Bainbridge and Bell (2001), Jones et al. (2008), and Rebelo et al. (2012). Jones et al. is particularly interesting because its lead author is the creator of SharpEye, one of the best-known commercial OMR programs, and the article contains a detailed description of how SharpEye works. It also has some insightful discussion of OMR evaluation, as does the paper by Rebelo et al.

3.1 Is evaluation really a serious problem yet?

A plausible argument can be made that OMR research/development is in such a primitive state that there is as yet no need for evaluation. It is claimed that, since OMR is still a young field, substantial advances can still be made without evaluation.[1] We do not dispute the claim, but it does not necessarily follow that evaluation is not yet a serious issue! A lack of standardized evaluation may be appropriate in a field in its *infancy*, where no one has tried to achieve the same thing as a previous worker but in a different way, but not in a *young* field: the OMR literature already contains many examples of problems that multiple people have tried to solve in multiple ways. In addition, commercial OMR systems have now been available for over 15 years. For most of that time, their vendors have made claims about their systems' accuracy that are not merely impractical to compare, but so ill-defined as to be virtually meaningless. Bellini et al. (2007) and Jones et al. (2008) both report commercial claims of accuracy around 90%, and outlandish claims of much greater accuracy appear from time to time. For instance, the German version of the product homepage for Capella Scan (retrieved 30 July 2013) claimed typical accuracy of 97%, and the homepage for PhotoScore Ultimate 7 (retrieved 30 April 2014) claimed no less than 99.5% 'on most PDFs and originals'. But, as noted by both Bellini et al. and Jones et al., it is not clear how any of the commercial systems measure accuracy, nor on what music; we have literally never seen an accuracy claim for a commercial system that answered either of these questions.

As an (admittedly unscientific) survey, we informally asked participants on the mailing list of the International Society for Music Information Retrieval and a few of our colleagues with more than 10 years experience with OMR tools about their perceptions of the weaknesses of current (as of 2014) OMR systems. Unsurprisingly, several answers provided anecdotal evidence that tools had improved over time, with later versions of tools from the larger commercial vendors considered to be performing well, but no systematic indication of the type or complexity of the scores on which tools perform well or poorly.

3.1.1 Competition between systems on accuracy

In sum, there has been no *effective* competition between OMR systems on the basis of accuracy, and – as a result – there is far less incentive than there might be for vendors to improve their products, and somewhat less incentive for researchers to come up with new ideas.

The only detailed comparison of OMR systems we know of was carried out by Craig Sapp of Stanford University (Sapp, 2013). Sapp tested SmartScore X2 Pro and SharpEye 2.68 with the first page of the first movement of Beethoven's Piano Sonata no. 1. In a previous study (Sapp, 2008) he tested only one program, SharpEye 2.63, with a page of a piano/vocal arrangement of the folksong *Farewell to Lochaber* and with the final page of Mozart's Piano Sonata no. 13, K. 333. The music test corpus the current paper proposes includes the Mozart as Test Page 21, and the Beethoven as Page 30. See Section 8 below. His analysis makes it clear that accuracy measures depend heavily on how errors are counted. One or both OMR systems have problems with grace notes and articulations (e.g. missing or moving staccatos), text, and fingerings and occasionally move or misplace slurs and dynamics. But the quality of the scan is quite high and the music is not that

[1] Indeed, we have encountered forceful statements from OMR researchers to this effect, e.g. 'I think plenty of progress can still be made on OMR without evaluation' (Chris Raphael, personal communication, January 2013).

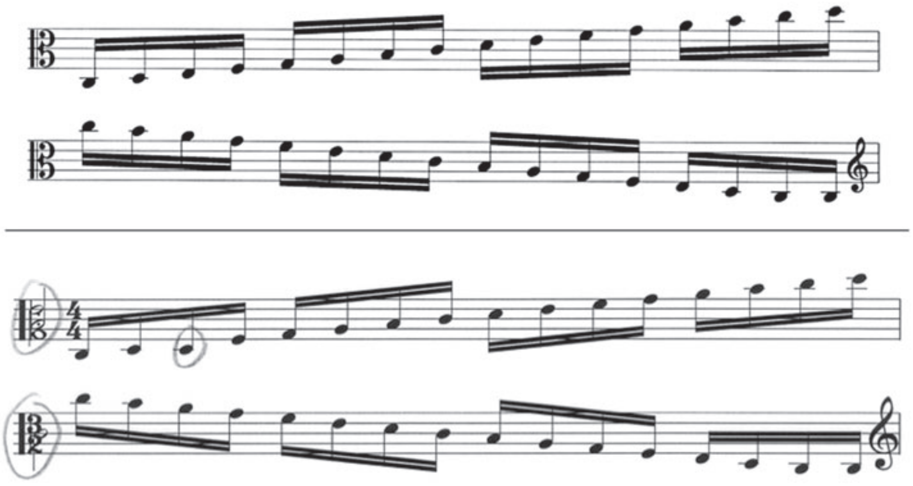

Fig. 1. Top: original 'born digital' image. Bottom: output of an OMR system.

complex, and both systems correctly identify the duration and especially the pitch of nearly all notes.

3.2 End-to-end versus subproblem evaluation

The obvious way to approach evaluation of any system is 'end-to-end', that is, to measure the performance of the system as a whole. But with a problem as difficult as OMR, there is much to be said for comparing ways to solve subproblems: in the case of OMR, tasks like identifying staff lines, segmenting the image into symbols, recognizing the individual symbols, and so on. Rebelo et al. (2012) lists available materials for evaluating a number of OMR subproblems.

Unfortunately, such stage-at-a-time evaluation is impossible to perform for nearly all commercial OMR systems because they are effectively 'black boxes', i.e. they afford access only to their final results. Also, stage-at-a-time evaluation assumes that stages are independent. Such independence certainly holds for nearly all OMR systems in use now, both commercial and other: in particular, the vast majority of existing OMR systems we are familiar with work in a strictly bottom-up manner. But in some of the most promising work on OMR now going back many years – McPherson (2002) built the first system of this type; Rossant and Bloch (2007), and Raphael and Wang (2011) are other examples – the stages can interact. (It is difficult to see how a strictly top-down system could function, and to our knowledge no one has attempted one.) Figure 1, showing an excerpt from the 'OMR Quick-Test' (Ng & Jones, 2003) as rendered by a long-gone version of a commercial OMR program, hints at the value of interacting stages: the C clef at the left end of each system has been replaced by a configuration of superimposed notes, barlines, and (on the lower staff) time signature that superficially resembles it.[2] The mistake is doubly ridiculous: first, the sets of superimposed musical symbols are nonsensical, and second, the overwhelming majority of staves in CWMN begin with a clef! A more flexible architecture can make it easy to avoid this kind of mistake. In McPherson's system, a module at a given level can object to information passed to it from a lower-level module and request a different interpretation.

Under the circumstances, we are most interested in the end-to-end approach, and that is what the remainder of this paper is concerned with.

4. OMR is difficult because CWMN is extremely complex

One of us has written previously (Byrd et al., 2010) that 'An appropriate evaluation metric [for OMR systems] is, to put it mildly, not straightforward'. The difficulty of evaluating OMR systems is an inevitable result of the complexity of CWMN. But why is CWMN so complex? One reason is that text and speech are both representations of natural language, and natural language has always been one-dimensional; but Western music notation represents Western music, which – except for monophonic music like unaccompanied songs – is fundamentally two-dimensional and graphical: music will usually contain several interleaved strands (e.g. instruments or voices), each containing notes having both pitch and duration. A second reason (in part a result of the first) is the enormously complex semantics of music notation and the high degree of context dependency in symbols that it entails.

Context and pitch notation. To see how context can affect pitch notation alone, consider three notes in a brief passage from Henry Purcell's *The Epicure* (Figure 2(a)). We are particularly interested in how the pitch of the third of these notes is affected by its context.

The standard pitch-notation rules of CWMN have been in effect for over 200 years now. As those rules are usually stated, an accidental – for example, the sharp just to the left of the first labelled note in the figure – affects the pitch of every following note on the same staff line or space, until the next barline. But the 'usual statement' of the rules is incomplete: the accidental

[2] The OMR program was a version of PhotoScore from about 2006.

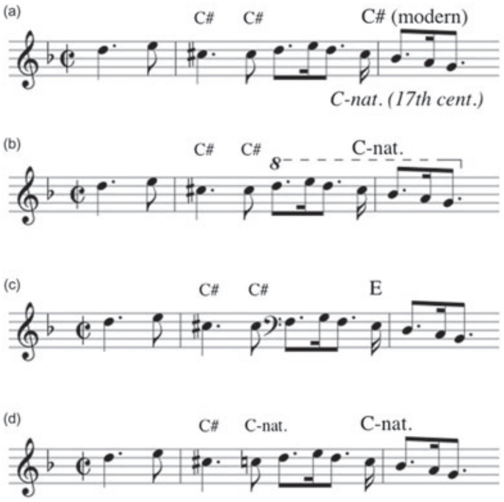

Fig. 2. (after Purcell). The third labelled note might be C♮, C♯, or E.

really affects only notes of the same diatonic pitch in the same octave and clef (Gould, 2011). Thus, in Figure 2(a), the music Purcell actually wrote, the third labelled note would be C♯. In (b), the third note is in a different octave, and in (c), it is in a different octave and is not even a C; neither is affected by the sharp on the first note. In (d), the second note is C in the same octave with a natural, so the third note is C♮. However, *The Epicure*, written in about 1692, antedates the modern rule, and by 17th-century rules, the third note in (a) is in fact C♮! This is to avoid an augmented interval between it and the following note; see Rastall (1982, p. 180).

To make the complexity of the situation in Figure 2(c) explicit, the pitch of the last marked note is (in ISO notation) E3, that is, E below 'middle C'. Seeing this requires taking into account (1) the *absence* of a sharp or flat for E in the key signature; (2) the *presence* of a change to bass clef affecting it; (3) the *absence* of accidentals for its pitch on preceding notes in its bar; and (4) the *absence* of an octave sign affecting it.

Context and time notation. The effects of context on CWMN notation of time – duration and rhythm – are no less dramatic than on pitch notation. For example, Figure 3 shows the first few bars of Schubert's Impromptu in E♭, D. 899 no. 2. Notice that, while there are no triplet markings after the first bar, the right hand (on the upper staff) has triplets throughout. That is clear because otherwise the nine eighth notes in each bar would greatly exceed the bar duration indicated by the 3/4 time signature and confirmed by the left-hand part. And this is a serious matter because if the (invisible) triplets are not represented somehow, notes in the two hands will not be synchronized. Nor is this a rare occurrence: unmarked tuplets – especially triplets – are quite common.[3]

CWMN is arguably the most sophisticated of any well-known notation in any field; it is certainly the most sophisticated we know of. Of course the concept of 'sophistication' is not well defined, but Byrd (1984, pp. 49–56) compares CWMN in some detail to two plausible alternatives, Chinese writing and mathematical notation. He concludes that Chinese is far simpler: like every known written form of natural language, it is one-dimensional, albeit with an extraordinarily complex and ill-defined character set. Mathematical notation is a worthier opponent, but Byrd lists four ways in which CWMN is more complex; we give an updated version of his discussion below.

4.1 Why CWMN is more complex than mathematical notation

In this section, we argue that music notation is more complex than mathematical notation in a general sense rather than just complex for recognition. However, we believe that almost everything we mention except the 'fit the notation to the page' considerations in issue #4 *can* make CWMN harder to recognize.

First, we need some unusual terminology. A *hunk* is a set of symbols on a single staff in CWMN that includes one or more notes, all beginning at the same time, and that, from a notational standpoint, behaves as a nearly inelastic unit. (The term was coined by Gomberg (1975). Unfortunately, it has not caught on, and there is no standard term for this concept, but it is closely related to the TEX concept of a 'box'. It is also closely related to one sense of the musical term *chord*, but chords are sometimes considered to extend across all staves. In addition, chords contain only notes; hunks contain symbols attached to the notes as well.)

A *system* is a set of staves containing music to be played synchronously. If two or more staves are involved, this is indicated by a curly brace or square bracket across all the staves.

For comparison to CWMN, Figures 4 and 5 are two typographically complex pieces of mathematical notation. CWMN is still more complex than mathematical notation for the following reasons:

(1) In mathematics, relative horizontal positioning of items is rather straightforward: indeed this requires *vertical* alignment within mathematical 'hunks' such as matrices or arrays, but otherwise horizontal lines are always totally independent. Achieving vertical alignment of the items – for example, numbers, factors, and terms – in the horizontal lines of a matrix or array of equations is trivial: every item in every line must be aligned with a corresponding item in each of the other lines. But in music, the relative horizontal positions of the hunks can be rather complex, for example when tuplets are involved. At the end of the page from Ives' *The Housatonic At Stockbridge* shown in Figure 6, tuplets of 3 notes

[3] The Schubert example and a number of other examples of the subtlety of real-world CWMN are shown and discussed in the online 'Gallery of Interesting Music Notation' (Byrd, 2013).

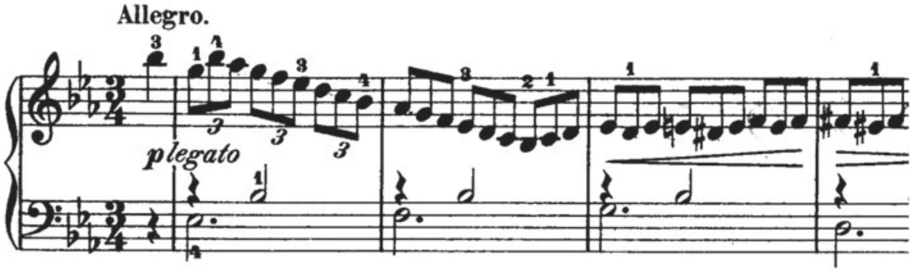

Fig. 3. (Schubert). The upper staff has triplets throughout.

in the time of 2; 5 in the time of 4; and 10, 11, and 14 in the time of 8 are in effect simultaneously with normal subdivisions of the beat. Thus, durations are being scaled by five different amounts at that point in the music. (It looks at first as if seven scale factors are involved, but 10:8 = 5:4, and the 'tuplet' marked 4 in the bassoon part – the fourth staff down – is really not a tuplet, but is nested inside a 3:2 triplet.) This leads to an intricate series of interleaved start times for the notes, and their relative horizontal positions should follow suit.

Whereas horizontal lines (except for hunks) are independent in mathematical notation, the same is not true in CWMN. In Figure 7 (from Brahms' Intermezzo, Op. 117 no. 1), on the last 8th note of the first bar in the top staff, the downstemmed note forces the upstemmed note to move to the left of its normal position. This type of interaction is not unusual; it can also be seen in the piano part of Figure 6 and in Figure 8. It happens in a more extreme form on the downbeat of the second bar, where notes in the chords in both staves already occupy no less than three horizontal positions; the 8th notes on both staves cannot share any of those positions, and must instead move even further to the right!

(2) Another factor is a question of character sets. Neither mathematics nor CWMN is based entirely on a predefined character set, but in several respects CWMN is much further from having such a basis. One of the non-predefined symbols in CWMN – the slur – appears more complex than anything we know of in mathematical notation. Slurs do not have a fixed size, shape, or orientation. About all that can be said is that a slur is more-or-less a smooth curve in the mathematical sense. And slurs that cross system breaks are an exception even to that: not the entire slur, but each section of such a slur is a separate smooth curve. In Figure 9, at least two slurs extend from the last bar of the first system to the to first bar of the second system. In Figure 10 (from *Scarbo*, the last piece in Ravel's *Gaspard de la Nuit*), two slurs start in the upper system and cross into the lower system; the beginnings and ends of both are numbered. Note also that each slur crosses between the staves of a system; slur #1 also crosses back. Figure 11 (from *Le Tombeau de Couperin*, also by Ravel) is a slur with seven inflection points.[4] A slur is also more-or-less a function of the horizontal coordinate, i.e. a vertical slice through a slur will nearly always intersect it at only one point, but they can back up for short distances. Cross-system slurs show another reason why CWMN is further from a fixed character set than is mathematics, namely *interruptibility*: it is not a special feature of slurs in CWMN, but is built into the system. In fact, almost any 'line-like' symbol in CWMN can be interrupted, for example a beam (Figure 8, from Debussy's *Danseuses de Delphes*, Durand edition).

(3) The last factor is related to the practical requirement of fitting the notation on pages. In this respect mathematical notation is less demanding than natural language, since individual pieces of mathematics nearly always fit on a page. But (except for short examples like the figures in this paper) CWMN is *more* demanding than natural language. It has all the line-justification, running header and footer, etc., requirements, plus two more: if possible, page turns must occur at a convenient point for someone playing a musical instrument, and the last page must be filled.

4.2 Complexity-related classification of CWMN

For purposes of OMR evaluation, we will find it useful to classify instances of CWMN because system A might be much more accurate than system B on one type of music, but system B might be far more accurate on another type. One obvious way to classify CWMN would be with categories defined by difficulty of recognition. Naturally, some music in notated

[4] The overall graphical context in Figure 10 – with intersecting beams and so on – is exceptionally complex. On the other hand, while the slurs in both Figures 10 and 11 are more complex than the average, they're nowhere near the most complex we know of. The record holder appears in the online 'Gallery of Interesting Music Notation' (Byrd, 2013); it has no fewer than 10 inflection points, spans three staves in each of three systems, and goes backwards several times.

$$
\begin{aligned}
G(z) = e^{\ln G(z)} &= \exp\left(\sum_{k\geq 1} \frac{S_k z^k}{k}\right) = \prod_{k\geq 1} e^{S_k z^k/k} \\
&= \left(1 + S_1 z + \frac{S_1^2 z^2}{2!} + \cdots\right)\left(1 + \frac{S_2 z^2}{2} + \frac{S_2^2 z^4}{2^2 2!} + \cdots\right)\cdots \\
&= \sum_{m\geq 0}\left(\sum_{\substack{k_1, k_2, \ldots, k_m \geq 0 \\ k_1 + 2k_2 + \cdots + mk_m = m}} \frac{S_1^{k_1}}{1^{k_1} k_1!}\,\frac{S_2^{k_2}}{2^{k_2} k_2!}\cdots\frac{s_m^{k_m}}{m^{k_m} k_m!}\right) z^m
\end{aligned}
$$

Fig. 4. Complex mathematical notation: algebraic manipulation of a generating function.

$$
\begin{array}{ccccccccc}
 & & a_{12}x^{(2)} & + & a_{13}x^{(3)} & + & a_{14}x^{(4)} & = & \lambda x^{(1)} \\
a_{21}x^{(1)} & & & + & a_{23}x^{(3)} & + & a_{24}x^{(4)} & = & \lambda x^{(2)} \\
a_{31}x^{(1)} & + & a_{32}x^{(2)} & & & + & a_{34}x^{(4)} & = & \lambda x^{(3)} \\
a_{41}x^{(1)} & + & a_{42}x^{(2)} & + & a_{43}x^{(3)} & & & = & \lambda x^{(4)}
\end{array}
$$

Fig. 5. Complex mathematical notation: an eigenvalue equation featuring a matrix with missing entries.

form is inherently much more difficult to recognize than other music, regardless of other factors (the quality of printing, of the scan, etc.). Byrd (2008) discusses this fact and gives examples of easy and very difficult music. However, reducing the difficulty of notated music to a one-dimensional scale in a meaningful way is no easy feat. We know of no published classification of music-notation complexity, but some attempts at distinguishing levels of notational complexity do exist in the literature. Byrd's MeTAMuSE multiple-recognizer OMR project (Byrd & Schindele, 2006, 2007) made an attempt at some distinctions appropriate for OMR. For example, they defined Complexity Level 2 – the second simplest of four – as follows:

> One voice per staff, but chords without 2nds or unisons are allowed; not-too-complex rhythm (no tuplets except marked triplets) and pitch notation (no octave signs). No lyrics, cues, or key-signature changes. Beams and slurs in a single system are allowed, but cannot cross staves. No cross-system anything (chords, beams, slurs, etc.). Common dynamics (*pp* to *ff*) and hairpins are allowed; trills, common articulation marks, and fingerings are allowed.

That definition does not say whether a large number of features of CWMN are allowed; besides, it would be easy to argue that many of the exact features it *does* specify should be allowed in Level 1, or not allowed until Level 3. This example suggests how difficult it is to devise a sensible scale of complexity.

We now have a scheme that does not take the idea of a total ordering as seriously, but which allows much easier and more well-defined classification. We describe it later (in Section 7.1), but an unusual concept that it relies on is worth mentioning here. First consider one more example, an excerpt from a Nocturne by Chopin written in idiomatic piano style (Figure 12). This example contains, among other complexities, instances of voices appearing and disappearing in the middles of measures; individual noteheads belonging to two voices; and interlocking beams. All of these features appear in many piano pieces by Chopin and other composers.

Five of the six examples of complex CWMN in this section are solo piano music. This is not an accident. Appendix C of this paper argues that the overwhelming majority of the most complex notation occurs in piano music, and explains why that should be the case. Accordingly, we have dubbed this type of CWMN *pianoform* notation. Pianoform notation is on multiple staves, one or more having multiple voices, and with significant interaction between and/or within staves; see Section 7.1.1 and Appendix C.

5. The main issues for OMR evaluation

Among others, Bainbridge and Bell (2001), Byrd et al. (2010), and Rebelo et al. (2012) have discussed the problems of devising evaluation methods for OMR systems. Byrd et al. argue that the main issues are:

(1) The level at which errors should be counted.
(2) Number of errors versus effort to correct.
(3) Relative importance of different symbols.
(4) Variability of complexity of the notation.

The first three make sense to us, but the fourth is problematic in that the complexity of the notation – important as it is – is only one of several factors that can make one page image far more challenging than another, regardless of the other issues. For example, Figure 13 shows part of a clean, well-printed

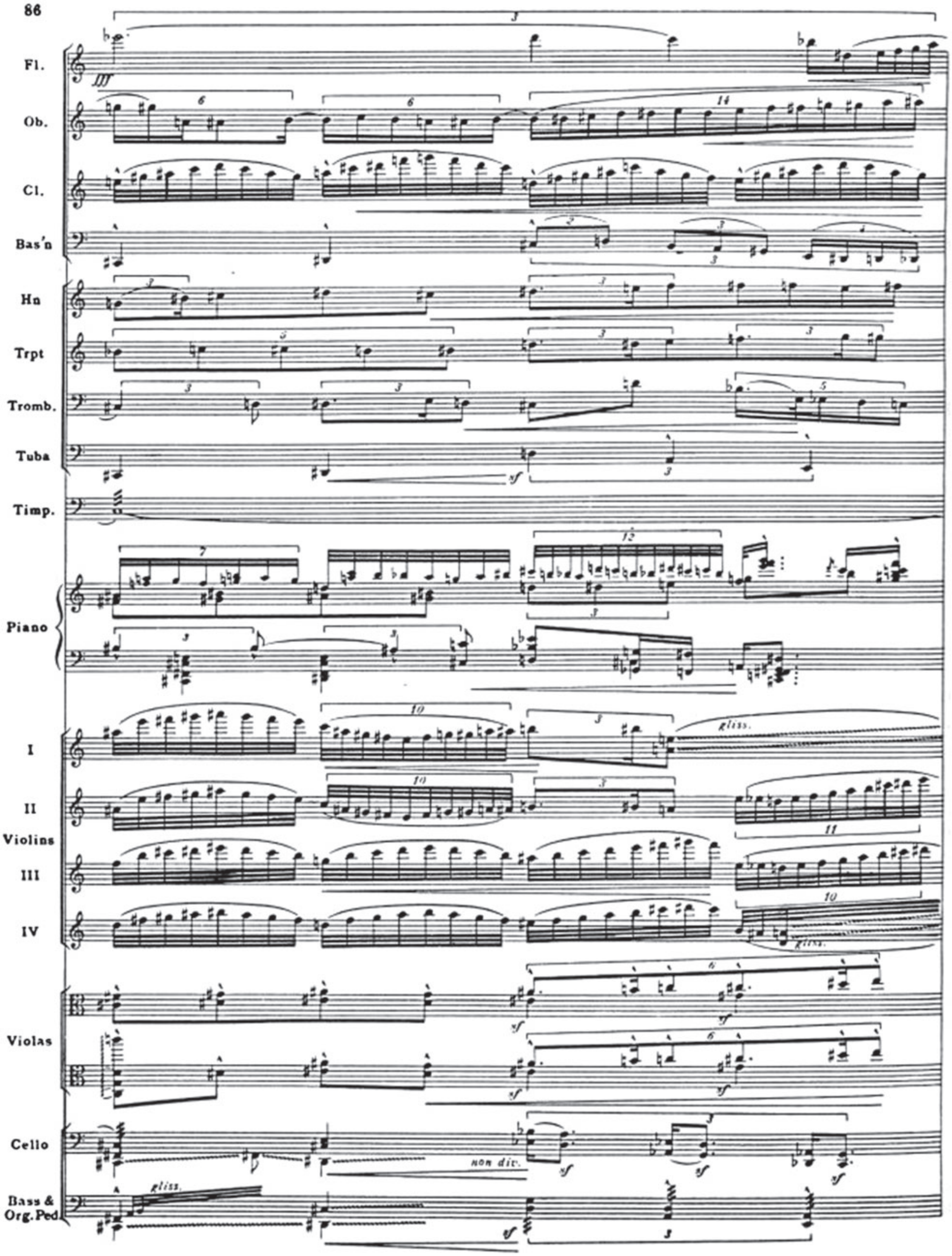

Fig. 6. (Ives). Simultaneous tuplets of many kinds.

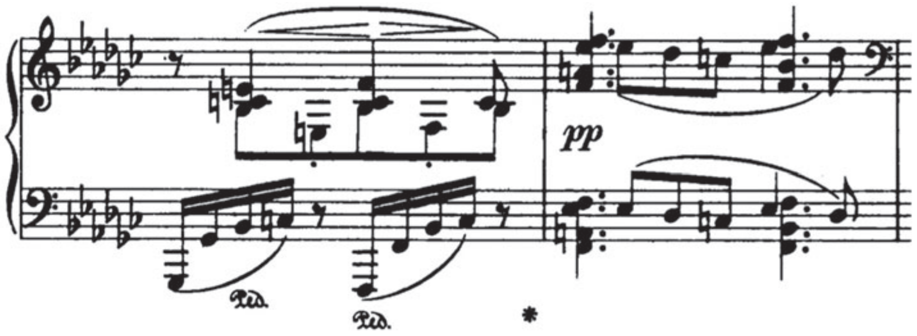

Fig. 7. (Brahms). Interaction between voices.

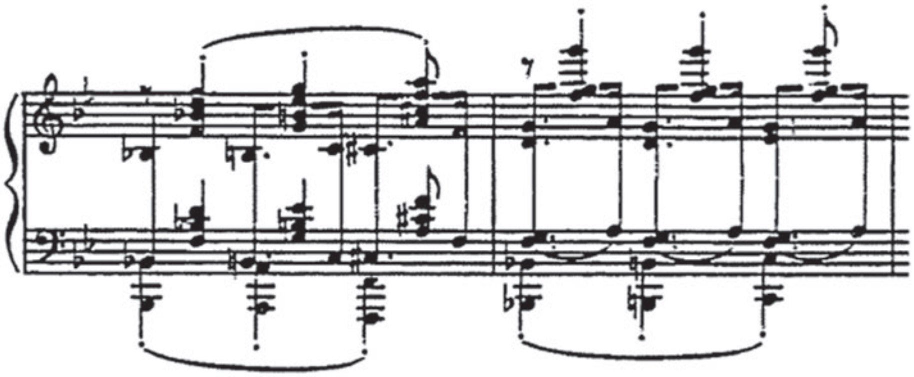

Fig. 8. (Debussy). Beams interrupted by chords.

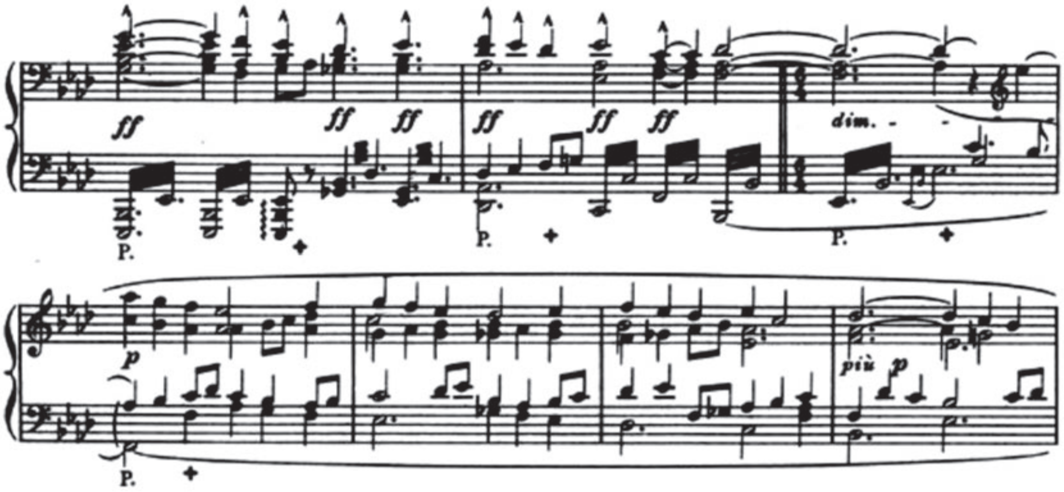

Fig. 9. (Wagner). Complex notation, difficult for OMR.

copy of a page of the cello part of a Haydn quartet, scanned from a nicely-engraved edition with the page edges almost perfectly aligned with the axes of the image (Test Page 14 in our corpus). Another image of the same page might have been scanned from a poor photocopy with numerous pencil markings (common in library-owned scores), rotated by six degrees. If so, recognizing the first image versus recognizing the second would be another instance of ‘comparing apples and oranges’. To include factors like these, we prefer to make the fourth problem *Variability of the page image*.

We now discuss these issues. See Bainbridge and Bell (2001), Byrd et al. (2010), and Rebelo et al. (2012) for more detail.

5.1 Issue 1. Level of evaluation

A phenomenon that is discussed more or less directly by Reed (1995), Bainbridge and Bell (2001), Droettboom and Fujinaga (2004), and others, is: *document-recognition systems can be described at different levels, and they may behave very*

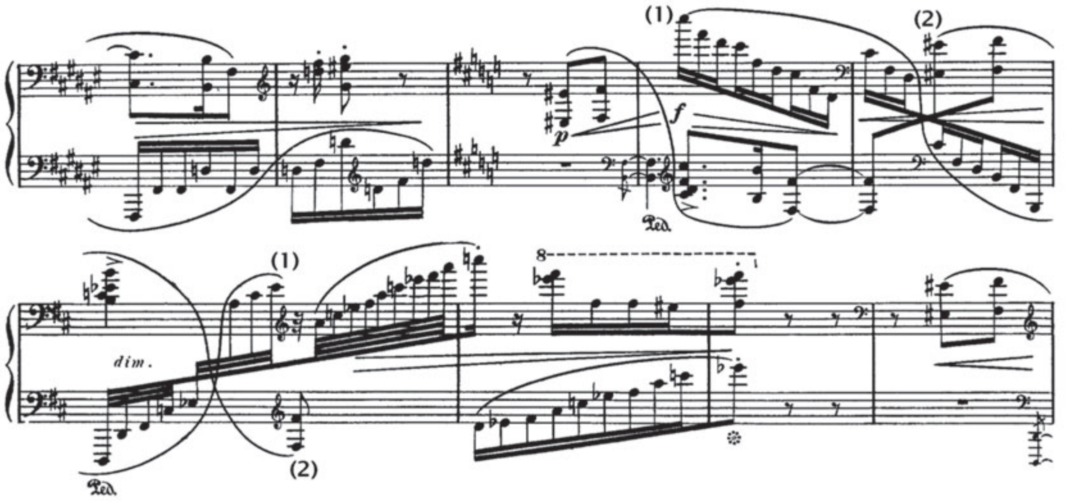

Fig. 10. (Ravel). Two cross-system, cross-staff slurs.

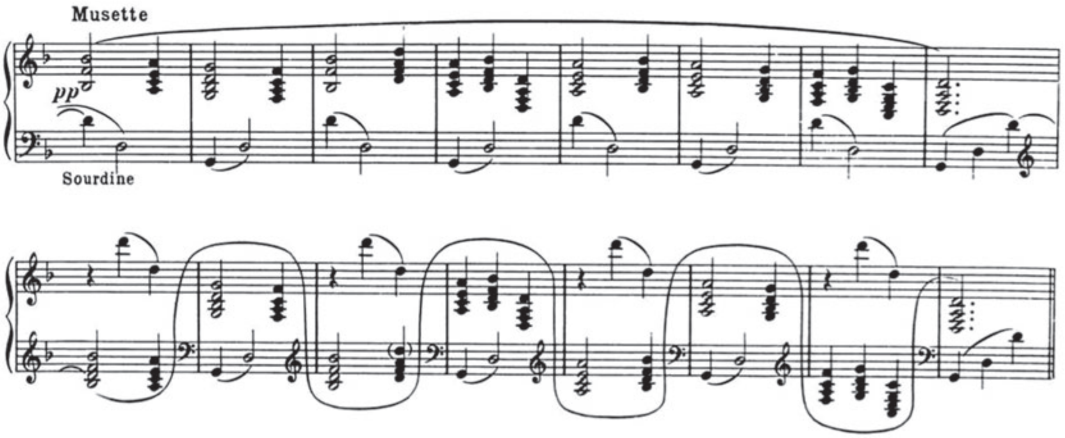

Fig. 11. (Ravel). A slur with seven inflection points.

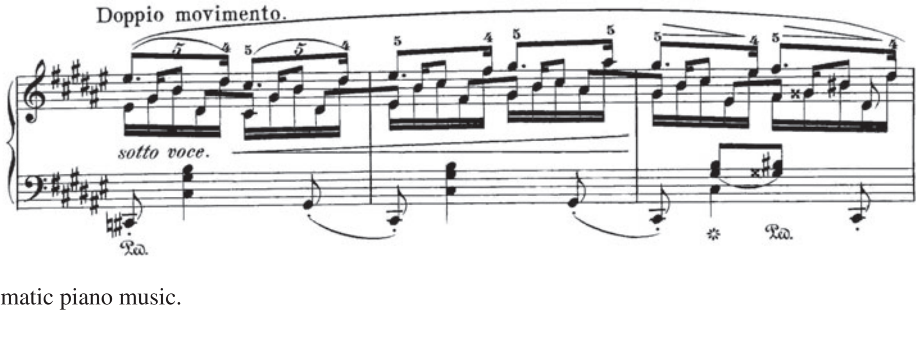

Fig. 12. (Chopin). Idiomatic piano music.

differently at each level. (The same is true of a great many complex systems. Hofstadter (1999) discusses this phenomenon in ant colonies, brains, chess boards, human bodies, computer programs, and several other systems.) In particular, document-recognition systems may be far more accurate at one level than at another. Any document page is a scanned image, and therefore the behaviour of almost any optical recognition system can be described at the pixel level – the lowest possible level; but such a description is not likely to be very informative. OCR systems are usually evaluated at the low level of characters, and for most text-recognition situations, that is satisfactory. If not, the word level – a higher level – nearly always suffices, and the relationship between the two levels is very straightforward. With music, however, things are much more complex.

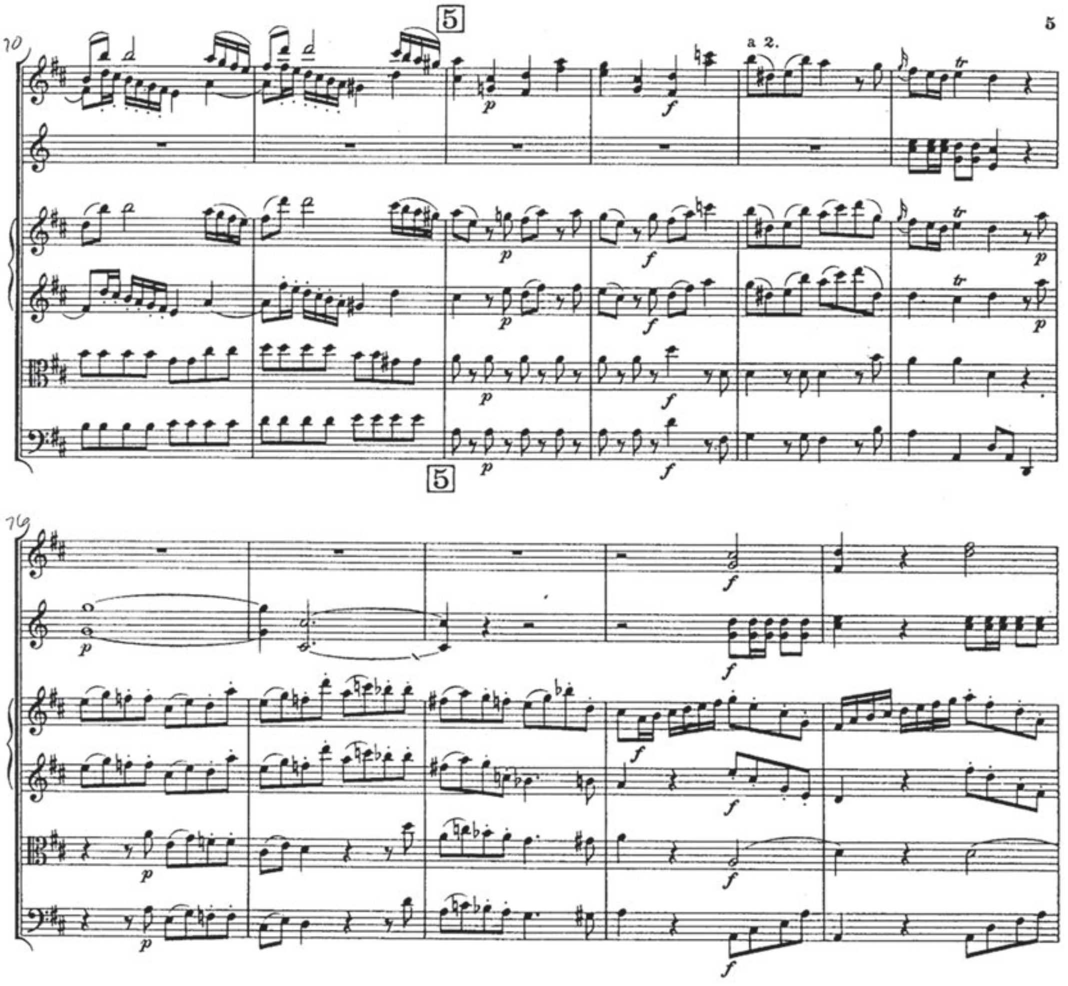

Fig. 13. (Haydn). A clean, well-printed, well-scanned and aligned page.

5.1.1 Levels of description of CWMN

Reed (1995, p. 72) distinguishes two important levels of description of CWMN: ‘One possible definition defines symbols in terms of the primitives used during recognition: beams, stems, note heads, flags, dots, etc. Alternatively, symbols may be defined in terms of musical constructs: half notes, quarter notes, eighth notes, rests, etc. The difference can be surprisingly significant when reporting recognition results’. Along the same lines, Bellini et al (2007) refer to *basic* symbols (graphic elements: noteheads, flags, the letter ‘*p*’, etc.; these have no meaning by themselves) and *composite* or *complete* symbols (eighth notes with flags, beamed eighth notes, chords, dynamic marks like ‘*pp*’ and ‘*mp*’, etc.).

There is only a relative handful of basic symbols, but a huge number of composite symbols. Droettboom and Fujinaga comment:

> In many classification problems the evaluation metric is fairly straightforward. For example at the character level of OCR, it is simply a matter of finding the ratio between correctly identified characters and the total number of characters. In other classification domains, this is not so simple, for example document segmentation, recognition of maps, mathematical equations, graphical drawings, and music scores. In these domains, there are often multiple correct output representations, which makes the problem of comparing a given output to high-level ground truth very difficult. In fact, it could be argued that a complete and robust system to evaluate OMR output would be almost as complex and error-prone as an OMR system itself. Symbol-level analysis may not be directly suitable for comparing commercial OMR products, because such systems are usually ‘black boxes’ that take an image as input and produce a score-level representation as output.

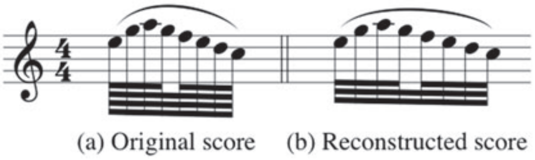
(a) Original score (b) Reconstructed score

Fig. 14. A minor problem at low level, but a serious problem at high level.

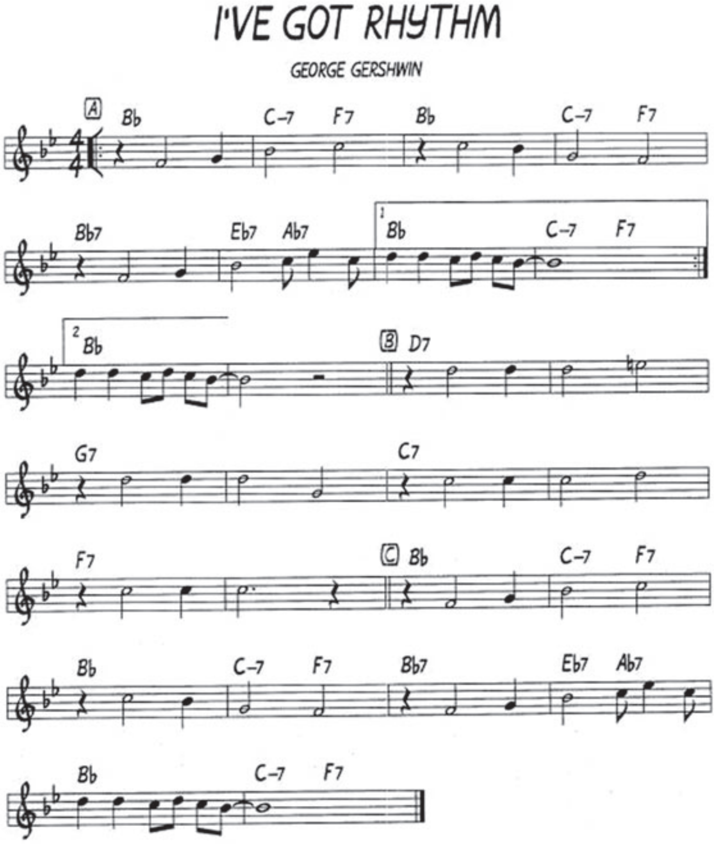

Fig. 15. (Gershwin). Lead sheet.

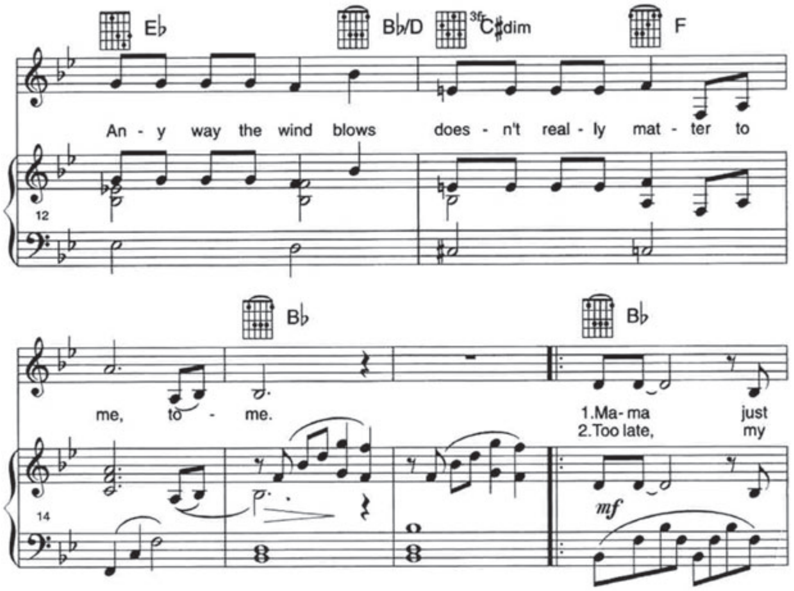

Fig. 16. (Mercury). Piano/vocal/guitar arrangement.

Note particularly the last statement; their 'symbols' are likely identical to Bellini et al's 'basic symbols', but their 'score-level representation' is clearly a level above Bellini et al's 'composite-symbol-level representation'. In fact, these problems of OMR directly reflect the intricate semantics of music notation. The last marked note in Figure 2(c) is a composite symbol consisting of notehead, stem, and flags. Its duration of a 16th note is clear just from the composite symbol. However,

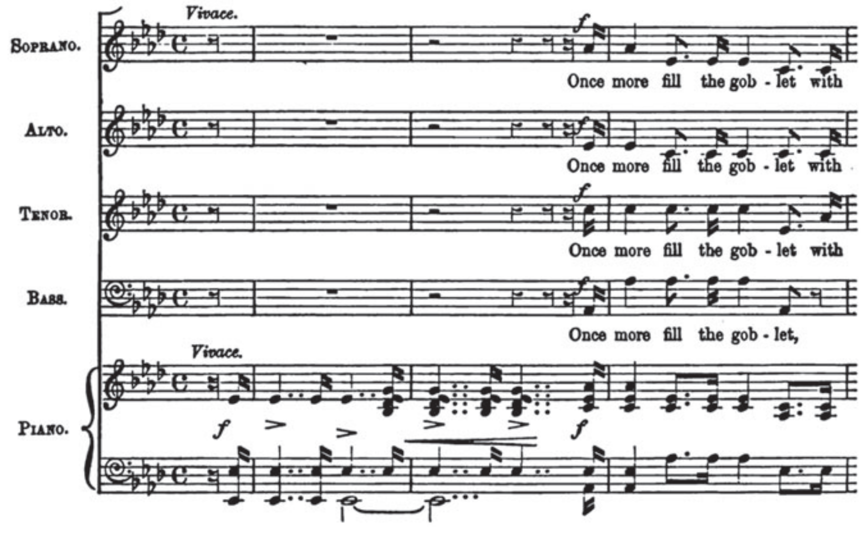
Vivace.
SOPRANO.
ALTO.
TENOR.
BASS.
PIANO.
Once more fill the gob - let with
Once more fill the gob - let with
Once more fill the gob - let with
Once more fill the gob - let,
f

Fig. 17. (Schumann). Music set from movable type.

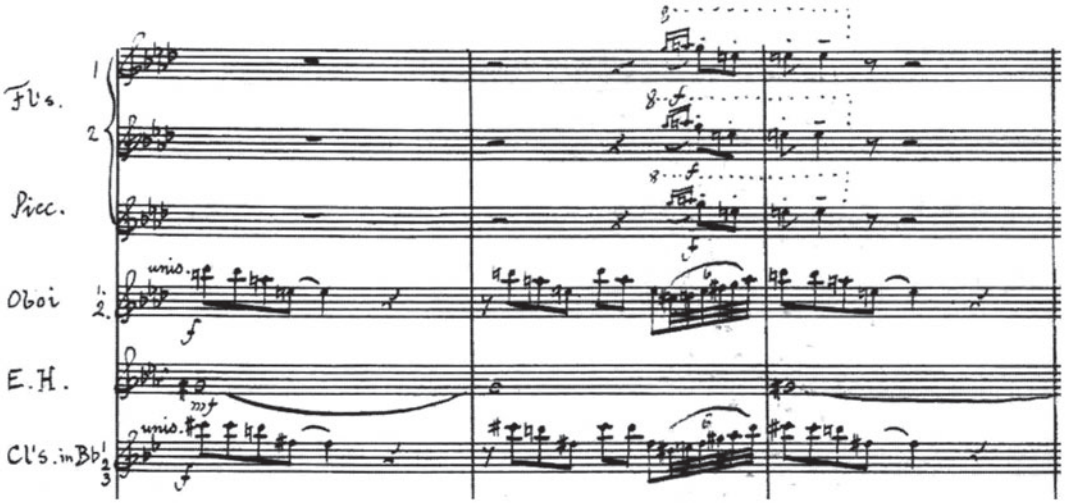

Fig. 18. (William Grant Still). Music autography.

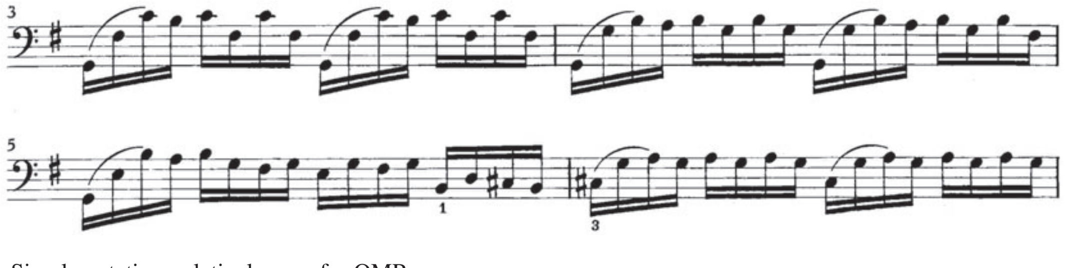

Fig. 19. (Bach). Simple notation, relatively easy for OMR.

seeing that its pitch is (in ISO notation) E3 requires taking into account the four factors described under 'Context and pitch notation' in Section 4 above.

Instead of 'basic' and 'composite' symbols, many authors have spoken of *low-level* symbols and *high-level* symbols, and we prefer the latter terminology. Table 1 lists the levels of description of CWMN that we have mentioned.

5.1.2 Levels of description and error rates

Now consider Figure 14 (from Reed, 1995, p. 73). In this case, the high-level symbols are the clef, time signature, notes – 64ths in Figure 14(a), 32nds in Figure 14(b) – and slur. Both the clef and the slur is a single low-level symbol. But the time signature and notes are comprised of multiple low-level symbols: for the former, digits; for the latter, noteheads and beams (and, in other cases, flags, accidentals, augmentation dots, etc.).

Reed points out that in this example (ignoring the clef and time signature), the only problem in the reconstructed score is a single missing beam, and if low-level symbols are counted, 19 of 20 (95%) are correct. But if you count high-level symbols instead, only 1 of 9 (11%) is correct: every note has the wrong duration. This example shows how a mistake in one low-level symbol can cause numerous errors in high-level symbols. But context dependence in music notation is not just a matter of low-level versus high-level symbols. For example, getting the clef wrong is likely to cause *secondary errors* in note pitches for many following bars: perhaps dozens of notes, if not more (we do not consider the problems in Figure 14 secondary errors because the missed beam is actually part of each note, both graphically and functionally).

Unfortunately, each level has its advantages (Droettboom & Fujinaga, 2004), so the best choice depends on the situation. Both of the current authors used high-level evaluation in our previous research simply because we were working on 'multiple-recognizer OMR': the idea was to 'triangulate' in the output of several commercial OMR systems to come up with a reading more accurate than any of them. However, we had access only to these black-box systems' score-level representations, sometimes in symbolic (MusicXML or other) form, sometimes only in graphical form. Under the circumstances, high-level evaluation made more sense.

5.2 Issue 2. Number of errors versus effort to correct

It is not clear whether an evaluation should consider the number of errors, perhaps weighted in some way, or the amount of work necessary to correct them. The latter is certainly more relevant for many purposes, but it is very dependent on the tools available, e.g. for correcting the pitches of notes resulting from a wrong clef. Bainbridge and Bell (2001) propose overcoming this objection by thinking in terms of a hypothetical editor designed for correcting OMR mistakes. This is an interesting idea, but it does not fully solve the problem. As Ichiro Fujinaga of McGill University has pointed out (personal communication, March 2007), the effort needed to correct errors also depends greatly on the distribution and details of the errors: it is far easier to correct 100 consecutive eighth notes that should all be 16ths, than to correct 100 eighth notes whose proper durations vary sprinkled throughout a score, a consideration also evident in some weighting schemes for error counting (e.g. Ng & Jones, 2003). A closely related issue is whether 'secondary errors' clearly resulting from an error earlier in the OMR process should be counted, or only primary errors?

5.3 Issue 3. Relative importance of different symbols

In recognition systems for media like text, at least for natural languages, it is usually reasonable to assume that all symbols and all mistakes in identifying them are equally important. In other cases, one might want to distinguish a very small number of symbol classes, perhaps two – one for independent characters and one for subsidiary symbols like diacritical marks – and a very small number of mistakes. (In French, for example, diacritical marks are important and fairly common, e.g. in words like *élève*, 'student', and *garçon*, 'boy'. In English, diacritical marks are quite rare and are invariably optional: e.g. the diaresis in *naïve* may be omitted at the discretion of the writer.) With music, nothing that simple is even remotely adequate. It seems clear that note durations and pitches are the most important things, but the relative importance ranking of other symbols is not obvious. Identifying the voice membership of notes (at least) is important in many cases but not all. How important are redundant or cautionary accidentals? Fingerings? Articulation marks?

Ichiro Fujinaga has pointed out (personal communication, March 2013) the value of weighting errors differently for different users and types of music. The analogy to natural-language text is worth revisiting. As compared to the roman alphabet of Western European languages, Polish adds a character, the crossed 'L', and Norwegian a crossed 'O'; the Cyrillic alphabet of Russian is almost completely different. How should one evaluate OCR systems that have varying degrees of accuracy on these characters? The answer depends on what language or languages one is interested in. Similarly, some symbols are common and important in some musical repertoires but rare or nonexistent in others. Compare Figure 15 (part of a 'lead sheet' containing the melody and chord symbols for Gershwin's 'I've Got Rhythm') and Figure 16 (from a piano/guitar/vocal arrangement of Freddie Mercury's 'Bohemian Rhapsody') to the other musical examples in this paper. Figure 15 contains 16 chord symbols, the short character strings like 'Bb' above the staves, and Figure 16 contains six, but none of our other figures includes any chord symbols. Figure 16 contains six chord frames, the little grids with dots in them, which show a guitarist where to put their fingers; none of the others contains a single one. And lyrics are a vital feature of Figures 16 and 17, but they do not occur in the other figures.

As we describe above, Bellini et al. (2007) advocate employing weights assigning relative importance to symbols, and they give a weighting based on a survey of OMR experts. Their work is a laudable attempt to add some objectivity to weighting OMR errors. However, we deem it almost certain that users interested in different types of music would assign weights to symbols very differently from these experts. We would like to see repertoire-specific lists of weights created in a similar fashion.

5.4 Issue 4. Variability of the page image

Before discussing the issues that arise from the variability of the page image, we must note that printed music in the public domain is of considerably more interest for OMR than music that is still protected by copyright, partly because a variety of

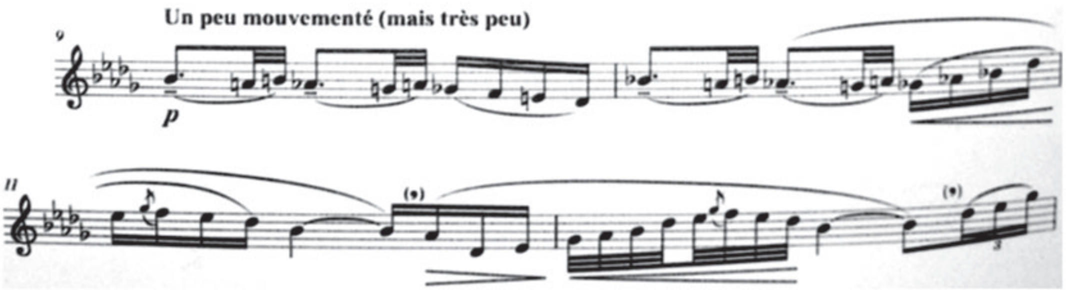

Fig. 20. (Debussy). Curved image.

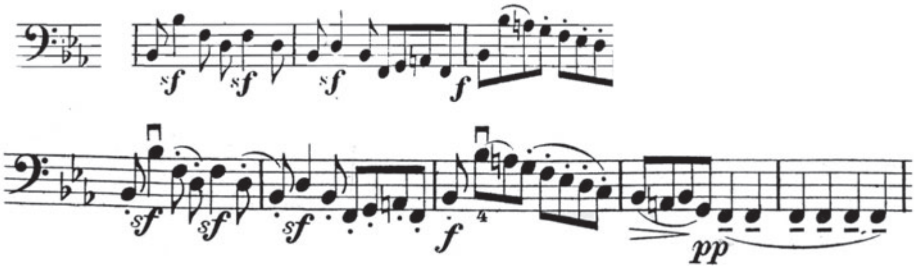

Fig. 21. (Beethoven). Tightness.

scans or replications is available free of charge to researchers (see the disclaimers on the IMSLP, imslp.org, website for evidence of this), but mostly because users can do anything they want with public-domain editions at no cost. Therefore, very old publications – say, from the 1920s or before – are much more important than one might expect.

5.4.1 Notation

The examples of 'real' music in the current paper – Figures 3, 6 to 13, 15, 17 to 21 – demonstrate the tremendous variability of CWMN (as opposed to its rendering on a physical page). The complexity described in Section 4 shows that notation is highly complex. The variability of the page image entails at least two more challenges, described below.

5.4.1.1 Style conventions: movable type (and autography) versus engraving. While late 20th-century and 21st-century computer-typeset printed music follows standard conventions with little graphical variation among editions except for fonts and spacing, this degree of graphical consistency is a very modern phenomenon. OMR systems must contend with printed music whose graphic makeup may differ considerably. One important factor is the method of typesetting. Until late in the 20th century, most published music was prepared on solid metal plates with a mixture of stamping (for fixed characters) and engraving (for symbols of indefinite size and/or shape like beams, slurs, and stems) (Ross, 1970); this is called *plate engraving*. But a significant amount was 'set' with two other methods: *autography* – essentially freehand pen and ink, often with stamps for fixed characters like clefs, accidentals, and noteheads – and *movable type*. The former is seen in Figure 18, in an excerpt from William Grant Still's *Afro-American Symphony*. This music was not prepared with stamps, and little needs to be said about it: the potential difficulty for an OMR system is obvious (and few OMR systems claim to handle hand-copied music). But movable type is another story.

Movable type is the technology Gutenberg is famous for. It is seen in Figure 17, from an 1885 edition of a Schumann song arranged for chorus and published by the well-known British firm Novello. Movable type is a clumsy technology for material as complex as typical CWMN is, and it was almost obsolete by the early 19th century until it was revived in a large number of publications from Novello, starting roughly in the 1840s (Russell, 1997). Novello used movable type for relatively simple music that was expected to sell many copies, for example hymnals. They took their leap backwards on economic grounds. Movable type is cast, so it can be made from a hard metal; but engraved plates are necessarily made from a rather soft alloy that does not stand up well to printing thousands of copies. Therefore, large print runs from engraved plates required re-engr aving, driving up the publisher's expenses considerably. The advent of photoengraving in the 20th century eliminated this disadvantage of engraving by making it possible to copy an engraved plate onto a surface of harder metal.

Music set from movable type almost always has frequent tiny breaks in staff lines and between noteheads and their stems, and simplified flags on notes. In addition, and for no obvious reason, Novello's movable-type publications have peculiarly-shaped bass clefs, quarter rests represented by backwards eighth rests, and other odd features. All of these features are visible here, and all can cause problems for OMR systems.

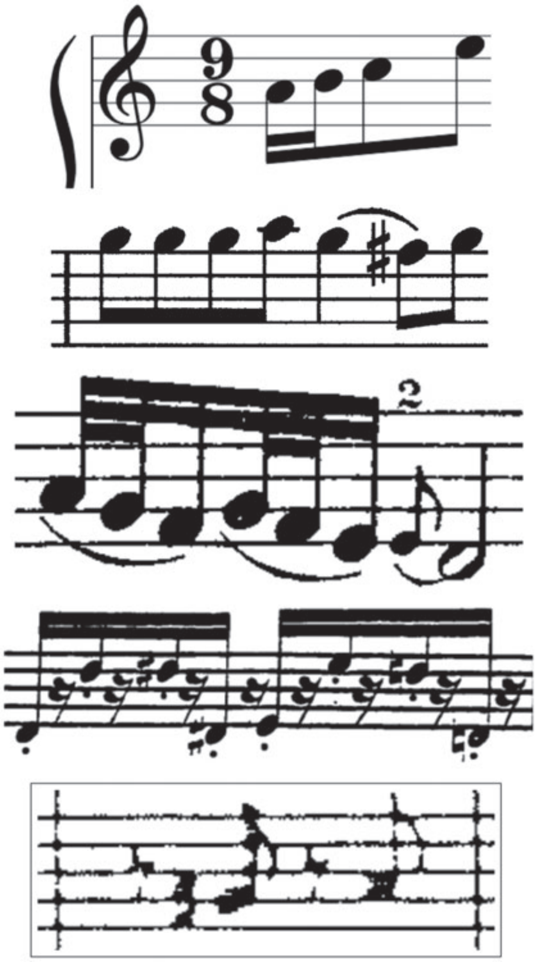

Fig. 22. Image quality grades 1 (top) through 5 (bottom).

5.4.1.2 The challenge of variable complexity. More serious than purely graphical variability, the complexity of notation of some music inherently presents far greater challenges for OMR than does the notation of other music, independent of the printing quality and condition of the page image (Byrd et al., 2010). Given the same quality of scanned images, it seems obvious that almost any conceivable OMR system will do better on a page of fiddle tunes or of music for solo cello (Bach, Cello Suite no. 1, Prelude, Barenreiter edition: Figure 19) than a page of Ives' orchestral work *The Housatonic At Stockbridge* (Figure 6), of a Chopin piano piece written in highly idiomatic style for the piano (Figure 12), or a page of what we have described as pianoform music like the piano reduction of the Prelude to Wagner's *Parsifal* (Figure 9). The problems the solo cello music presents for OMR are a rather small subset of the problems each of the others present. Again, this factor hardly applies to text.

Unfortunately, properly taking notational complexity into account is extremely difficult, as we have shown in Sections 2 and 4 above. To our knowledge, no one has ever published a basis for rating the difficulty or complexity of CWMN for any purpose. We believe that there is a good reason for this omission: CWMN is replete with subtleties on every level, from shapes of individual symbols (e.g. slurs) to relationships between pairs of adjacent symbols (say, a notehead and a preceding accidental) to highly contextual features (e.g. a clef and key signature and a note several bars later). As a result, finding a principled way to define the complexity of a given piece of CWMN, for OMR or any other purpose, would be a major research project in itself. See the discussion of CWMN in Section 4 above and Byrd (1984). Nonetheless, the complexity of the notation obviously affects the performance of OMR systems so much that it makes no sense to ignore it.

5.4.2 Image quality

The image quality of the scanned music may vary considerably. Some scans are pristine high-resolution images of beautifully printed originals. Others have missing pixels in symbols, smudges, or other imperfections, due either to the quality of the physical original (e.g. poor printing, frayed edges or tears) or to human intervention (e.g. multiple generations of photocopying; careless scanning practices like failure to align the axes of the music with those of the scanner or accepting a curved image; or pen or pencil markings on the page). In Figure 22, compare the Grade 1 through 3 images with Grades 4 and 5. It would hardly be surprising to find OMR systems performing substantially better on images like the first levels than on images like the last ones, even if they contained the same music.

5.4.3 Tightness of spacing

Proximity between noncontiguous symbols is an important aspect of CWMN, serving invariably as a clue, and sometimes as an essential indication, of structural aspects of the music. An important case is in distinguishing the meaning of the small dots that pepper many pages. Are they augmentation dots, each increasing the logical duration of the note they belong to? Or are they staccato dots, with no effect on the *logical* duration of the associated note or notes, but affecting their *performed* duration? Most of the time, there is no problem, but compare Figure 21(a), a few bars from the bottom of a page of the cello part of the Beethoven Trio, Op. 3 no. 1, first movement, in the Schott edition, to Figure 21(b), the same music in the Peters/Herrmann edition (Test Page 15 of our corpus). Figure 21(b) is more tightly set than Figure 21(a); that is, there is less space between symbols in Figure 21(b). In particular, compare spacing of the last six eighth notes in the third bar of the excerpt. In the study by Byrd et al. (2010), one OMR program mistook each of the staccato dots in that bar for an augmentation dot on the preceding note! (It did not make that mistake with Figure 15(a).)

This is a mistake a competent human reader of the music would never make, regardless of spacing. People certainly depend on proximity to infer some structural aspects, but they use additional means to decide what is related to what: higher-level means that are very difficult to turn into an algorithm.

Looking again at Figure 21(b), for example, a long series of dotted notes is a rare and unlikely phenomenon. It is made still less likely by the facts these notes are shorter than quarter notes, that this music is by Beethoven, and that the bar these notes appear in is surrounded by bars with very common rhythm patterns. Besides, most of these dots are not *quite* in the right vertical position to be augmentation dots on the preceding notes. For a person who has been reading complex CWMN for years, these pressures on perception are so strong that it would likely never even occur to them that the dots might be augmentation dots. There are thus two pitfalls for OMR systems in Figure 21(b): (i) tightness makes it difficult to determine which symbols should be associated with which, because the cue of simple proximity is less reliable than usual, and (ii) it's hard to know which information needs to be taken into account: usually the location of other notes does not need to be taken into account when deciding if a dot is an augmentation dot or not, but in this case it does. Indeed, OMR suffers from the classic 'frame problem' of 'Artificial Intelligence' (McCarthy & Hayes, 1969), that it is very difficult to define a 'frame' to *ex*clude irrelevant information and *in*clude all relevant information for the correct solution of the problem. Given the state of Artificial Intelligence, it is likely that computers cannot do this in the forseeable future.[5] As a result, it is almost certain that for many years to come, tightly-set music will remain considerably more difficult for an OMR program than music with generous spacing.

Tightness is just one example of the general phenomenon that the more crowded the music is, the harder it will be for an OMR program to make sense of the notation. For example, vertical crowding could result in notes belonging to one staff being assigned to another staff, at least if their stems are pointing away from the centre of the staff: that happens routinely with two voices on a staff. But crowding can be caused by factors other than horizontal spacing of notes or vertical spacing of staves. However, tightness is the easiest to measure; it is also the only one we have actually observed leading an OMR program to make mistakes.

It is worth noting that, in printed music, augmentation dots are often larger than staccato dots, but by no means always; compare Gould (2011, p. 116) to Ross (1970, p. 169). In any case, the difference is slight, and it is doubtful whether it plays much of a role in disambiguating for either humans or computers.

6. Pros and cons of a standard OMR testbed

It is not hard to see some reason why a proper OMR testbed does not currently exist. Ideally, any such testbed would include a large collection of test pages – say, at least a few hundred pages, to cover all combinations of symbols and score qualities – and would be updated regularly in order to keep systems from being targeted at a known, static dataset, in a fashion similar to the current practice at TREC and MIREX. But resources are scarce: the OMR community is much smaller than the text IR, speech recognition, and musical audio communities, and there is no commercial organization representing the companies involved in OMR. We doubt that an annually updated testbed with a lot of music, associated tasks, and impartial evaluation is forthcoming anytime soon.

On the other hand, almost every new OMR project we have seen uses a different evaluation method and test collection, apparently put together from scratch. Besides the obvious duplication of effort, it seems unlikely that most of the resulting testing systems are well planned – and, of course, comparing the performance of different systems is virtually impossible. Maintaining an ideal testbed (with a regularly-changing collection of test pages) may not be viable, a publicly available testbed with a static collection would at least let researchers compare their own results to those of others. Furthermore, even in the absence of active updates and curation, researchers could point out deficiencies in such a testbed that may be taken into account by future OMR projects utilizing it and perhaps overcome at some point.

After all, once evaluation metrics and a methodology for choosing test pages are agreed on, replacing the pages with new ones is a relatively simple matter. And if researchers or developers of OMR systems overfit their tools to the testbed's initial collection (the software equivalent of school teachers 'teaching to the test'), a new set of pages will stop them dead.

7. The way forward

As stepping stones towards establishing a standard testbed, we offer:

- definitions of categories for notation complexity and grades for image quality and tightness (to handle Problem 4);
- an evaluation metric in the form of OMR error types and guidelines for counting errors at a high level (to handle Problems 2 and 3); and
- a small corpus: 34 pages of music with varying grades of complexity, image quality, and 'tightness', chosen according to the guidelines in Appendix B.

These are all the components a real testbed requires, though, as explained above, no static corpus can be completely satisfactory. We believe that the *definitions* and *metrics* will be of value independently from the corpus. On the other hand, the definitions and corpus will be valuable independently of the metrics: for example, with a different set of metrics to support low-level evaluation (Problem 1).

We expect that a proper testbed will always require human (as opposed to automatic) sorting of examples and scans into appropriate categories. Indeed, for scans of higher complexity, the appropriate difficult tasks faced by OMR systems are present in the definition of the various categories (e.g. in

[5] Factors like this are the basic reason that one of us has argued (Byrd, 1984, pp. 5–6) that really high-quality formatting of complex music is a serious artificial-intelligence problem. See also Byrd (1994).

Table 2. MeTAMuSE error types.

	Description of Error
1	Wrong pitch of note (even if due to extra or missing accidentals)
2	Wrong duration of note (even if due to extra or missing augmentation dots)
3	Misinterpretation (other symbol for note, note for other symbol, misspelled line of text, slur beginning/ending on wrong notes, note stem interpreted as barline, etc.)
4	Missing note (even if due to note stem interpreted as barline)
5	Missing symbol other than notes (and accidentals and augmentation dots)
6	Extra symbol (other than accidentals and augmentation dots)
7	Gross misinterpretation (e.g. entire staff missing); *or*, note interpreted as cue or grace note

Section 7.1.3, categorizing scans by level of *tightness* requires that all noteheads can be identified).

7.1 Definitions and metrics

7.1.1 Complexity of the notation

As we noted in Section 4.2, reducing the complexity of notated music to a one-dimensional scale in a meaningful way is 'no easy feat', and Byrd's MeTAMuSE OMR project (Byrd & Schindele, 2006, 2007) made an attempt at distinctions appropriate for OMR, but was not very successful. We now describe a scheme that does not take the idea of a total ordering as seriously, but which offers categories that are both more well defined and easier to identify. These new categories might be described as *notated textures*. The categories are:

1. Music on one staff, strictly monophonic, i.e. one note at a time
2. Music on one staff, polyphonic, i.e. with chords or multiple simultaneous voices
3. Music on multiple staves, but each staff is strictly monophonic, with no *structural interaction* between them
4. *Pianoform* music: music on multiple staves, typically with multiple voices per staff, and with significant structural interaction, as is commonly seen in piano music (see below).

For the reasons discussed in Sections 4.2 and 5.4.2, the first category is undoubtedly the easiest music for OMR, the last the most difficult; the middle two are probably roughly comparable. The problems the first category presents for OMR are a fairly limited subset of the problems each of the others present, so it is clearly the easiest music for OMR. Similarly, as Section 4.2 and Appendix C suggest, the problems pianoform music presents are a large superset of the others, so it is clearly the most difficult. We believe that the second and third categories are of roughly comparable difficulty.

By 'structural interaction' we mean the kind of interaction that can complicate recognition, and the main issue is not merely spatial constraints. For example, a melody line moving across staves can complicate things this way, and it very likely will do so if a slur or beam or tuplet crosses staves with it. Interaction in Figure 10 involves slurs and beams crossing staves; in Figure 8, chords split between staves; and in Figure 9, a slur crossing a system break as well as staves.

By 'pianoform' music we mean music on multiple staves, with at least one staff having multiple voices, and with significant structural interaction between staves and/or between voices of a single staff. Figures 7 to 12 are all for piano and are all examples of the last category. For instance, Figure 10 to a great extent, and Figures 8 and 11 to a lesser extent, have interaction between staves. Figures 7 and 10* have structural interaction leading to three or more horizontal positions for the same logical time, and Figures 6 and 12 have voices appearing and disappearing at arbitrary points within a measure. For further clarification of and rationale for the notion of pianoform music, see Appendix C.

7.1.2 Image quality

Following the recommendations of Riley and Fujinaga (2003), we assume that all images are gray scale unless colour is used to convey important information: e.g. red noteheads to call attention to a hidden subject. As Riley and Fujinaga note, purely black-and-white scanning may fail to capture all pertinent information (e.g. space between beams may be blackened due to grayscale 'whitespace' being interpreted as black).

Our image-quality rating scale assumes inspection by naked eye with normal (20/20) vision. Presumably very low resolution images will automatically result in low grades (see comments on jaggedness in the descriptions below). Note that these ratings describe only the quality of the reproduction of the original page; they do not take into account the quality of the notation on that page. But even for reproduction quality, the ratings are not comprehensive: they do not consider factors like curvature (Figure 20) or rotation of the image. However, in our experience, visible curvature is quite rare: this is to be expected, since flatbed scanners – the most common type – generally attempt to press the page flat. The second is fairly rare, and in any case it can be corrected with software before attempting OMR if the image is in high resolution and/or gray scale.

Given these exclusions, we use five levels of quality that, to us, seem easily distinguishable by human inspection (Figure 22).

Table 3. OMR test corpus.

Test page no.	Notated Texture	Cmplx. Grade	Tightness	Image quality grade	Title	Catalogue or other no.	Publ. date	Display page no.	Edition or Source
1	1-M	2x	1	1	OMR Quick-Test	1	Cr.2003	*	IMN Web site
2	1-M	2x	1	1	OMR Quick-Test	2	Cr.2003	*	IMN Web site
3	1-M	3x	1	1	OMR Quick-Test	3	Cr.2003	*	IMN Web site
4	1-M	2x	1	1	Level1OMRTest1	-	Cr.2005	*	DAB using Ngale
5	1-M	2x	1	1	AltoClefAndTie	-	Cr.2005	*	MS using Finale
6	1-M	2x	1	1	CourtesyAccsAndKSCancels	-	Cr.2005	*	MS using Finale
7	1-M	2	1	3	Bach: Cello Suite no.1 in G, Prelude	BWV 1007	1950	4	Barenreiter/Wenzinger
8	1-M	2	1	2	Bach: Cello Suite no.1 in G, Prelude	BWV 1007	1967	2	Breitkopf/Klengel
9	1-M	2	1	3	Bach: Cello Suite no.3 in C, Prelude	BWV 1009	1950	16	Barenreiter/Wenzinger
10	1-M	2	1	3	Bach: Cello Suite no.3 in C, Prelude	BWV 1009	1967	14	Breitkopf/Klengel
11	1-M	2	1	2	Bach: Violin Partita no. 2 in d, Gigue	BWV 1004	1981	53	Schott/Szeryng
12	1-M	3	1	2	Telemann: Flute Fantasia no. 7 in D, Alla francese	TWV 40 #7	1969	14	Schirmer/Moyse
13	1-M	3	1	2	Haydn: Qtet Op. 71 #3, Menuet, viola part	H.III:71	1978	7	Doblinger
14	1-M	3	1	2	Haydn: Qtet Op. 76 #5, I, cello part	H.III:79	1984	2	Doblinger
15	1-M	3	3	3	Beethoven: Trio, I, cello part	Op. 3 #1	1950-65	3	Peters/Herrmann
16	1-M	3	1	2	Schumann: *Fantasiestücke*, clarinet part	Op. 73	1986	3	Henle
17	1-M	2	3	3	Mozart: Quartet for Flute & Strings in D, I, flute	K. 285	1954	9	Peters
18	1-M	3	1	2	Mozart: Quartet for Flute & Strings in A, I, cello	K. 298	1954	10	Peters
19	1-P	3	1	3	Bach: Cello Suite no.1 in G, Prelude	BWV 1007	1879	59/pt	Bach Gesellschaft
20	1-M	2	1	3	Bach: Cello Suite no.3 in C, Prelude	BWV 1009	1879	68/pt	Bach Gesellschaft
21	PF	3	1	2	Mozart: Piano Sonata no. 13 in Bb, I	K. 333	1915	177	Durand/Saint-Saens
22	PF	4	1	2	Ravel: Sonatine for Piano, I	-	1905	1	Durand
23	PF	4	1	2	Ravel: Sonatine for Piano, I	-	1905	2	Durand
24	2-P	3x–4x	1	1	QuestionableSymbols	-	Cr. 2005	1	MS using Finale
25	1-M	2x	1	2	Bellini #2, modif. from Pozzoli	1	1973	*	Ricordi
26	1-M	2x	2	2	Bellini #3, modif. from Gentilucci	3	1983	*	Curci
27	1-M	2x	1	2	Bellini #4, modif. from Curci	4		*	Curci
28	1-M	3x	2	2	Bellini #6, modif. from Pozzoli	6	1973	*	Ricordi
29	1-M	2x	1	2	Bellini #7, modif. from Gentilucci	7	1983	*	Curci
30	PF	3	2	3	Beethoven: Piano Sonata no. 1, I	Op. 2 #1	1915	2	Durand / Dukas
31	1-P	4	1	3	Bach: Violin Partita no. 2 in d, Chaconne	BWV 1004	1879	32	Bach Gesellschaft
32	1-P	4	1	2	Sor: Les Folies d'Espagne	Op. 15	1824	1	Meissonnier
33	4-M	3	1	3	Mozart: Quartet for Flute & Strings in D, III	K. 285	1882	7	Breitkopf
34	4-M	3	1	3	Beethoven: String Quartet #11	Op. 95	1863	9	Breitkopf

Notated Texture (discussed in Section 4.2):
1-M = Music on one staff, strictly monophonic (one note at a time)
1-P = Music on one staff, 'polyphonic' (with two or more voices)
n-M = Music on n staves (n > 1), but each is strictly monophonic, with no interaction between them
PF = 'Pianoform' music: music on multiple staves, one or more having multiple voices, and with significant interaction between and/or within staves
Cmplx. grade: complexity of the music, as defined by Byrd and Schindele (2007). 'x' suffixed to the level means the page contains artificial music, 'composed' or modified specifically for testing.
Image quality grade: graphical quality of the bitmap reproduction of the original page, as defined above (higher number = more flawed).
Tightness: how close horizontal spacing between notes is: 1 = adequate spacing throughout, 2 = some tight spacing, 3 = much tight spacing (more than 25% of notes are tightly spaced with other notes).
Publication date: Cr. = creation date for unpublished items.
Display page no.: the printed page number in the original. '/pt' means only part of the page.
Edition or source: IMN = Interactive Music Network; DAB = Donald A. Byrd; Ngale = Nightingale music-notation editor.

Grade 1: Near-flawless quality; expected from 'born digital' (as opposed to scanned) images. Absolutely no visual noise except for small isolated areas of grey (not black) pixels, no pixels turned on outside music symbols, no pixels turned off inside symbols. Each symbol (including staff lines, barlines, etc.) appears totally contiguous with essentially no jagged or fuzzy edges. Axes of music perfectly aligned with image axes: staves horizontal, with no skewing or curvature.

Grade 2: Very high quality; expected from the best scans. Small amounts of visual noise, no pixels turned off inside music symbols; a few pixels may be turned on outside symbols. Symbols may appear jagged, but each symbol is totally contiguous. Axes of music aligned with image axes: staves horizontal, with little or no skewing or curvature.

Grade 3: Expected from most high-quality scans. Some amount of visual noise, pixels may be turned on outside music symbols, and pixels may be turned off inside music symbols if this does not render the symbols noncontiguous (pixels may be turned off in note heads, inside dynamics or accidentals, but no barlines, stems, articulations, etc. may be broken). Axes of music reasonably well aligned with image axes.

Grade 4: Expected from many low-quality scans, including all scans of pages defaced with numerous pen or pencil markings or with serious bleedthrough into the image from another page. Visual noise sufficient to blur separation of symbols (e.g. accidentals or flags being amalgamated into staff or barlines). Pixels turned off may render symbols non-contiguous.

Grade 5: Expected from low-quality scans of defaced and/or poorly-printed originals. Large amounts of visual noise. Pixels turned on or off make sight reading difficult, and detailed inspection is necessary to identify some symbols.

7.1.3 Tightness of spacing

As we have pointed out, proximity between noncontiguous symbols is an important indication of structural aspects of the music. So the relevant question is just whether horizontal spacing between symbols is sufficient to avoid ambiguity – for example, between a staccato dot on one note and an augmentation dot on the preceding note when their vertical positions have a certain relationship. Compare the last six notes in the third bar of Figure 21(a), the Schott edition, and of Figure 21(b), the Peters/Herrmann, of the same music by Beethoven. Therefore we distinguish only three grades of tightness, 'adequate spacing throughout', 'some tight spacing', and 'much tight spacing'. For simplicity, we consider only spacing between noteheads.

In Figure 21(a), the horizontal space between these noteheads is about 0.6 times the 'interline space', the space between staff lines. In Figure 21(b), the space is only about 0.3 of the interline space, less than half as much. We define spacing as tight if the minimum space between any noteheads not part of the same chord is strictly less than 0.5 of an interline space; otherwise it is adequate.

7.2 How to count errors

7.2.1 Types of errors

Is it worth classifying errors into different types? Some researchers do, some do not, and, to our knowledge, there is no agreement on types among those that use them. We have ourselves used very different approaches in our previous work. Simonsen and his colleagues (Bugge et al., 2011) used a simple approach, reporting only total high-level errors. Byrd's MeTAMuSE multiple-recognizer OMR project (Byrd & Schindele, 2006, 2007) had a much more complex method, also counting high-level errors, but distinguishing seven types. Unless otherwise indicated, the word 'note' here means a normal note, not a grace note, cue note, or rest (Table 2).

We know of only one other well-documented taxonomy of OMR error types, that of Bellini et al. (2007). They distinguish just three types: for each symbol that is not correct, the error might be *miss* (the symbol present in the score was ignored), *add* (a symbol not present in the score was added), or *fault* (the symbol present in the score was not ignored, but was not identified correctly in some way).

It seems to us that distinguishing multiple types of errors is very desirable, as it may greatly facilitate evaluating systems for different purposes: for a given application, some errors might be much more or less important than others. But distinguishing types has at least two drawbacks. First, it inevitably takes more time; and second, in some situations it is hard to tell what type applies, if any. The next section discusses the latter problem.

7.2.2 Rules and guidelines for counting errors

The mere descriptions of error types above leave many questions unanswered: questions such as what to do with errors in symbols that really carry no information, that is, symbols that should be redundant. For example, we have seen cases where – despite the fact that there are no key signature changes on the whole page – an OMR program puts the wrong signature on just one system in the middle of the page. Additionally, as we discussed under Issue 3 above, errors should really be weighted differently for different users and types of music. But defining rules for counting errors in OMR is not a simple question; neither is customizing them for a given application.

As a starting point, the Appendix gives the rules and guidelines that the MeTAMuSE used for its high-level evaluation (Byrd et al., 2010). It should be noted that these rules were designed for classical music in general, and were created to help persons counting OMR errors manually, and mostly by looking at printed scores reconstructed from OMR. Another set of rules appears in Bellini et al. (2007) in conjunction with their taxonomy of error types mentioned above.

To our knowledge, no one has followed up the suggestion by Bainbridge and Bell (2001) (discussed above) that one should base OMR evaluation on the effort needed to correct the mistakes rather than on the number of mistakes, but ignoring secondary errors (defined above), as the MeTAMuSE rules do, is a step in that direction(Byrd et al., 2010) is not given in the reference section.

8. A small corpus of music pages

As a starting point for a test collection, we offer a small corpus comprising images of pages that we have used independently in prior work, augmented with pages mostly used by other researchers. The music pages described in Table 3 consist of the 'SFStudy' test corpus used in Byrd and Schindele (2006, 2007) (pages 1 through 24 in the table), including one also used by Sapp (2008); five of the seven test pages of Bellini et al. (2007), also used by Jones et al. (2008); another page used by Sapp (2013); and four additional pages. (Corpus B of Bugge et al. (2011)) is a slight variation of the SFStudy corpus, omitting test page 23 but adding two earlier versions of page 24. (The inclusion of the earlier versions of page 24 was unintentional.) Riley and Fujnaga (2003) recommend as 'best practices' that, in order to capture the smallest details in musical scores, they should be scanned with 8 bits of grey scale at 600 dpi. However, they note that 'print size varies between publications', with miniature scores having the smallest size, and add that 'for larger printed notation, 300 dpi may be sufficient'. These recommendations, which are now 12 years old, seem on the conservative side; in addition, none of the pages in our test collection are from miniature scores, and none have tiny details. And higher resolution strongly implies greater resources in terms of processing time as well as storage bandwidth. We believe that 300 dpi will be sufficient for OMR, and we offer the corpus at that resolution. We also offer most of the images at 600 dpi as well (a few were not available at that resolution). Images of the pages with those specifications can be downloaded (see Supplemental data).

The table itself is based on Table 1 of Byrd and Schindele (2007), adding image quality flaws according to the ratings of Bugge et al., but with higher numbers indicating more flaws (in Bugge et al., they indicate fewer flaws), plus new ratings of tightness of spacing as discussed above.

The 24 SFStudy pages were chosen in a principled way, based on the factors described in Appendix B (a minor revision of Byrd, 2008). Of these, TP1 through TP6 and TP24 are really catalogues of music symbols rather than examples of real music.

The Bellini et al. test pages are entirely monophonic. We omit two of their pages because they, as well as two pages we do include, have some unusual features that we do not want to over-represent and thereby bias evaluations.[6]

We emphasize that this corpus can only be a starting point for a serious test collection. For one thing, it contains no pages with image quality below our grade 3. But regardless of its content, 34 pages are not nearly enough to test the huge variety of situations an OMR program should handle.

8.1 A likely resource for future testbeds: IMSLP

The advent of IMSLP, the International Music Score Library Project, presents an interesting opportunity for OMR evaluation. As their Web site (imslp.org) says, IMSLP's Petrucci Music Library is 'a community-built library of public domain sheet music' containing an 'extensive collection of original scores scanned to PDF'. The collection is indeed extensive: as of November 2014, it claims to contain 295,000 scores or parts for 86,000 works. Thus, IMSLP has great promise as a source of material, both for OMR evaluation and for practical use in converting music to symbolic form via OMR. In fact, we obtained several of our test pages from IMSLP.

Not surprisingly, the scans vary considerably in quality. IMLSP formerly displayed ratings of image quality for its offerings, but, regrettably, it no longer does so. Instead, it shows ratings of one to five stars, apparently consensus ratings by users. But are these users rating the image quality, the edition of the music, or the music itself? We have not even been able to find any information as to what these ratings are *intended* to mean.

Clearly, it would be ideal if image quality ratings were supplied by trusted users adhering to a set of fixed, clearly described principles, and preferably augmented with ratings describing the complexity of the scores. But even without such ratings, we believe that IMSLP will prove to be a valuable resource.

9. Conclusions and future work

In December 2011, one of us (Byrd) received a message from the president of the company that develops and markets one of the best-known commercial OMR systems, arguing that comments about their company's product in Byrd et al. (2010) were unfair, and claiming that their system is the most accurate available and that its accuracy has improved tremendously over the years. This is obviously an important question for the company, and the aggrieved president may well be correct; *but there is at present no way to know*! This unfortunate situation need not and should not be allowed to persist. We believe that the advent of a real testbed – even one with a small, static collection of music pages which is updated infrequently – would substantially facilitate advances in the state of the art. As we have pointed out, the existence of such a testbed is not enough. It must be accompanied by a specification of how to count errors, of the level(s) at which errors are counted, and decisions must be made about whether certain errors should be weighted differently from others, and what the ratio of 'complex scores' to 'simple scores' should be; indeed, the definition of 'complex score' is itself very non-trivial. Furthermore, there are good reasons why such testbeds, and their accompanying specifications, have not materialized yet: for the reasons we have discussed, music OMR and its evaluation is far more difficult than OCR for most texts.

The obvious environment in which to employ an OMR testbed is MIREX and its tests held in conjunction with ISMIR every year. While the 'well-thought-out test collections and evaluation metrics' materials such a track needs do not really exist, we believe the materials and ideas the current paper presents constitute a significant step in that direction.

[6] Specifically, each of their pages 1, 2, 5, and 6 has multiple instances of C and F clefs appearing in positions that have been obsolete since the late 18th century. In addition, on these pages, stems on beamed 16th notes and shorter – i.e. notes with two or more beams – in the middle of a beamed group stop at the closest beam, while in the vast majority of published music, stems extend to the furthest beam. But the latter is much less likely to lead to OMR errors.

Even if 'ideal' testbeds and widely accepted metrics for OMR evaluation come into existence some day, challenges for evaluation will remain. For automated recognition of mathematics notation, Zanibbi and his collaborators have suggested (Zanibbi et al., 2011; Zanibbi & Blostein, 2011) using a bipartite graph layout to simultaneously capture errors in segmentation, classification and layout in expressions. Using such an approach in OMR is promising, one reason being that it is specifically tailored to take into account spatial and semantic relationships between symbols, not just their individual presence and correctness. As we have pointed out, mathematical expressions are considerably more rigidly structured than music notation: indeed, expressions are, in a very natural fashion, just operator trees over some signature. Nonetheless, there are clear similarities, e.g. accidentals, beams and dots are in obvious, and highly rigid, relationship with the notes they are associated to. It would be interesting to pursue an approach like Zanibbi's for the evaluation of OMR tools.

Finally, the labour-intensive manual error counting needed – the cost of which grows linearly with the size of a test collection – is an aspect of OMR evaluation that remains unaddressed. The work of Knopke and Byrd (2007) and Szwoch (2008) discussed earlier is an interesting step towards a solution that deserves following up. Another possible way to alleviate this problem would be to include scores in the testbed that have been automatically generated to exhibit phenomena that are known to be hard for OMR systems to handle. In order not to skew performance statistics, the frequency of notational challenges in such automatically generated scores would need to reflect prevalent frequencies in actual music under consideration, which would admittedly be genre (and composer) dependent. Similarly, a better selection of scans of varying image quality may be produced by artificially degrading superior scans. This approach has been used successfully in the document-recognition community at large; it has the great advantage that multiple tests have the same ground truth.

Acknowledgements

The definitions of quality levels and images illustrating them are reprinted with permission of our collaborators Brian Søborg Mathiasen, Esben Paul Bugge, and Kim Lundsteen Juncher. We use test pages from the corpus of Bellini et al. (2010) with permission of the authors.

Ichiro Fujinaga's comments on both early and late versions of this paper were extremely helpful, as were numerous conversations with Chris Raphael over a period of years. Dorothea Blostein offered her expertise on document-recognition systems in general as well as on music-recognition systems. Bill Clemmons and Craig Sapp offered comments based on many years of experience using OMR systems. Finally (in a literal as well as a figurative sense), extensive feedback from the referees greatly improved the paper. Most of Donald Byrd's work on OMR was part of the MeTAMuSE project, made possible by the generous support of the Andrew W. Mellon Foundation. Bill Guerin and Megan Schindele made substantial contributions to MeTAMuSE.

Disclosure statement

The authors certify that they have no financial interest or benefit arising from direct applications of the research reported in this manuscript.

Supplemental data

Supplemental data for this article can be accessed at https://doi.org/10.1080/09298215.2015.1045424.

ORCID

Jakob Grue Simonsen http://orcid.org/0000-0002-3488-9392

References

Bainbridge, D., & Bell, T. (2001). The challenge of optical music recognition. *Computers and the Humanities, 35*(2), 95–121.

Bellini, P., Bruno, I., & Nesi, P. (2007). Assessing Optical Music Recognition Tools. *Computer Music Journal, 31*(1), 68–93.

Bellini, P., Bruno, I., & Nesi, P. (2003, rev. 2014). *Assessment of optical music recognition tools, OMR assessment (type 1)* Working Paper. http://www.disit.org/5932

Bugge, E.P., Juncher, K.L., Mathiasen, B.S., & Simonsen, J.G. (2011). Using sequence alignment and voting to improve optical music recognition from multiple recognizers. *Proceedings of the 12th International Society for Music Information Retrieval Conference (ISMIR 2011)* (pp. 405–410). Canada: International Society for Music Information Retrieval.

Byrd, D. (1984). (Doctoral dissertation, Computer Science Dept., Indiana University). Ann Arbor, Michigan: UMI ProQuest (order no. 8506091); also available from www.npcimaging.com. Retrieved (in scanned form). Music notation by computer. http://www.informatics.indiana.edu/donbyrd/Papers/DonDissScanned.pdf

Byrd, D. (1994). Music-notation software and intelligence. *Computer Music Journal*, *18*(1), 17–20. Available (in scanned form) at http://www.informatics.indiana.edu/donbyrd/Papers/MusNotSoftware+Intelligence.pdf

Byrd, D. (2008). Guidelines for choosing OMR test pages (and factors affecting OMR accuracy). MeTAMuSE/IU Project Working Paper. Retrieved from http://www.informatics.indiana.edu/donbyrd/MROMR2010Pap/

Byrd, D. (2013). Gallery of interesting music notation. Retrieved from http://www.informatics.indiana.edu/donbyrd/InterestingMusicNotation.html

Byrd, D., & Schindele, M. (2006). Prospects for improving OMR with multiple recognizers. *Proceedings of the 7th International Conference on Music Information Retrieval (ISMIR 2006)* (pp. 41–46). Canada: University of Victoria.

Byrd, D., & Schindele, M. (2007). Prospects for improving OMR with multiple recognizers, revised and expanded version. Retrieved from http://www.informatics.indiana.edu/donbyrd/MROMRPap

Byrd, D., Guerin, W., Schindele, M., & Knopke, I. (2010). OMR evaluation and prospects for improved OMR via multiple recognizers. Retrieved from http://www.informatics.indiana.edu/donbyrd/MROMR2010Pap/OMREvaluation+Prospects4MROMR.doc

Cleverdon, C. (1967). The Cranfield tests on index language devices. In K. Sparck Jones & P. Willett (Eds.), (1997) *Readings in information retrieval*. San Francisco: Morgan Kaufmann.

Droettboom, M., & Fujinaga, I. (2004). Micro-level groundtruthing environment for OMR. In *Proceedings of the 5th International Conference on Music Information Retrieval (ISMIR 2004)* (pp. 497–500). Barcelona: Universitat Pompeu Fabra.

Gomberg, D. (1975). (Doctoral dissertation, Computer Science Dept., Washington University) A Computer-Oriented System for Music Printing, Ann Arbor, Michigan: UMI ProQuest.

Gould, E. (2011). *Behind bars*. London: Faber Music.

Hofstadter, D. (1979; twentieth-anniversary edition, 1999). *Gödel, Escher, Bach: An eternal golden braid*. New York: Basic Books.

Hook, J. (2011). How to perform impossible rhythms. *Music Theory Online*, *17*(4), article 1.

Jones, G., Ong, B., Bruno, I., & Ng, K. (2008). Optical music imaging: music document digitisation, recognition, evaluation, and restoration. *Interactive multimedia music technologies* (pp. 50–79). Hershey: IGI Global.

Kanai, J., Nartker, T.A., Rice, S.V., & Nagy, G. (1993). Performance metrics for document understanding systems. *Proceedings of the 2nd International Conference on Document Analysis and Recognition* (pp. 424–427). Piscataway, NJ: IEEE.

Knopke, I., & Byrd, D. (2007). Towards MusicDiff: A foundation for improved optical music recognition using multiple recognizers. In *Proceedings of the 8th International Conference on Music Information Retrieval (ISMIR 2007)* (pp. 123–126). Vienna: Austrian Computer Society (OCG).

McCarthy, J., & Hayes, P. J. (1969). Some philosophical problems from the standpoint of artificial intelligence. *Machine Intelligence, 4*, 463–502.

McPherson, J. (2002). Introducing feedback into an optical music recognition system. *Proceedings of the 3rd International Conference on Music Information Retrieval (ISMIR 2002)* (pp. 259–260). Paris: IRCAM Centre Pompidou.

Nagy, G. (1995). Document image analysis: Automated performance evaluation. In A. K. Spitz & A. Dengel (Eds.), *Document analysis systems* (pp. 137–156). Singapore: World Scientific.

Ng, K.C., & Jones, A. (2003). *A quick-test for optical music recognition systems*. Paper presented at the 2nd MUSICNETWORK Open Workshop, Workshop on Optical Music Recognition System, Leeds, September 2003.

Raphael, C., & Wang, J. (2011). New approaches to optical music recognition. *Proceedings of the 12th International Society for Music Information Retrieval Conference (ISMIR 2011)* (pp. 305–310). Canada: International Society for Music Information Retrieval.

Rebelo, A., Fujinaga, I., Paszkiewicz, F., Marcal, A.R.S., Guedes, C., & Cardoso, J.S. (2012). Optical music recognition: state-of-the-art and open issues. *International Journal of Multimedia Information Retrieval*, *1*(3), 173–190.

Reed, K.T. (1995). *Optical music recognition*, (MSc thesis). Dept. of Computer Science, University of Calgary, Canada.

Riley, J., & Fujinaga, I. (2003). Recommended best practices for digital image capture of musical scores. *OCLC Systems & Services*, *19*(2), 62–69.

Ross, T. (1970). *The art of music engraving and processing*. Miami: Hansen.

Rossant, F., & Bloch, I. (2007). Robust and adaptive OMR system including fuzzy modeling, fusion of musical rules, and possible error detection. *EURASIP Journal on Advances in Signal Processing, 2007*, article ID 81541.

Russell, D. (1997). *Popular music in England 1840–1914: A social history*. Manchester University Press.

Sapp, C. (2008). SharpEye examples: Mozart Piano Sonata No. 13 in B♭ major, K 333 and *Farewell to Lochaber*. Retrieved from http://craig.sapp.org/omr/sharpeye

Sapp, C. (2013). OMR comparison of SmartScore and SharpEye. Retrieved from https://ccrma.stanford.edu/~craig/mro-compare-beethoven

Selfridge-Field, E., Carter, N., & McGee, W. (1994). Optical recognition: A survey of current work; An interactive system; Recognition problems; The issue of practicality. In W. Hewlett & E. Selfridge-Field (Eds.), *Computing in Musicology*, Vol. 9 (pp. 107–166). Cambridge, MA: MIT Press.

Szwoch, M. (2008). Using MusicXML to evaluate accuracy of OMR systems. *Proceedings of the 5th international Conference on Diagrammatic Representation and Inference* (LNCS 5223, pp. 419–422). Berlin: Springer-Verlag.

Zanibbi, R., & Blostein, D. (2011). Recognition and retrieval of mathematical expressions. *International Journal on Document Analysis and Recognition, 15*(4), 331–357.

Zanibbi, R., Pillay, A., Mouchère, H., Viard-Gaudin, C., & Blostein, D. (2011). Stroke-based performance metrics for handwritten mathematical expressions. *Proceedings of the International Conference on Document Analysis and Recognition, ICDAR 2011* (pp. 334–338). Piscataway, NJ: IEEE.

Appendix A. MeTAMuSE rules and guidelines for counting errors

The MeTAMuSE project used the following rules and guidelines for its high-level evaluation (Byrd & Guerin, 2007). It should be noted that these rules were created to help persons counting OMR errors manually, and mostly by looking at printed scores reconstructed from OMR.

For clarity, we have reworded some of the rules and many of the 'rationales' listed.

(1) Differences from the original created by the program that printed the OMR'd music but that the OMRprocess is clearly not responsible for should

be ignored. Examples we have seen include which measures have measure numbers; adding a default instrument name like ‘Nylon Guitar’; font changes; and even rhythm changes, the latter caused by Finale spacing things so that notes that don’t fit the duration the measure ‘should’ have appear to be in the following measure. Less clear-cut cases include secondary errors resulting from other errors, like anticipatory key signature changes for beginning-of-system ‘key signature changes’ invented by the OMR program.

(2) Count errors even in symbols that really carry no information, hence should be redundant. For example, we have seen cases where – despite the fact that there are no key signature changes on the whole page – an OMR program gets the wrong signature in just one system in the middle of the page: that error should be counted.

(3) Separate graphical symbols that don’t correspond to anything in the original: count as ‘extra element’, even if they aren’t clearly recognizable as any musical symbol. This applies, for example, to little squiggles superimposed on beams that the OMR program might have intended to be slurs.

(4) Dotted slurs in the original: ignore; also ignore anything that evidently results from them, e.g. staccato or augmentation dots. Rationale: we assume that originals should not contain any dotted/dashed slurs. This is reasonable because recognizing curved dotted/dashed lines seems to be beyond the state of the art of optical pattern recognition.

(5) Missed (or added) fingerings: ignore rather than counting as missing symbols (or extra elements). Rationale: for most purposes, fingerings are not very important; we could count them as, say, 1/20 as much as a missing note, but it’s not worth the effort.

(6) Missed (or added) accents, articulation marks, etc.: ignore rather than counting as missing symbols (or extra elements). Rationale: for most purposes, they aren’t very important; we could count them as, say, 1/10 as much as a missing note, but it’s not worth the effort.

(7) Key signature only partly correct: count as a misinterpreted symbol. For example, a key signature of three flats interpreted as one flat and two sharps is just one misinterpreted symbol; likewise if it’s interpreted as five flats and three sharps (unlikely though that is).

(8) Text string only partly correct: count as misinterpreted symbol. This applies to cases of extra, missing, and wrong characters in the string. In multi-line blocks, consider each line as a separate string.

(9) Note pitch is wrong for any reason, including missing or extra accidentals: count as pitch error only. Do not count these situations as missing symbols or extra elements. *Exceptions*: (a) if the pitch is wrong because of a missing or extra octave sign, count the octave sign itself as a missing symbol or extra element; do not count the pitch as wrong for the affected notes. (b) If a missing or extra accidental results in several following notes having the wrong pitch, count only the first note as having wrong pitch. If a missing or extra accidental results in *no* notes having wrong pitch, ignore it. (c) If a missing, extra or wrong clef results in several following notes having the wrong pitch, just count the clef as a misinterpreted symbol; do not count note pitch errors. Rationale for the exceptions: these are very likely secondary errors; in addition, for the ‘ignore’ part of (b), this is not important enough to bother with in any situation we can foresee.

10. Note duration is wrong for any reason, including missing or extra augmentation dots: count it as a duration error and nothing else. In particular, do not count as missing symbols or extra elements. *Exceptions*: if the duration is wrong because of a missing or extra tuplet sign, count the tuplet sign itself as a missing symbol or extra element; do not count the duration as wrong for the affected notes. Rationale for the exceptions: these are very likely secondary errors.

(11) Missing extender (dashed line) following text: ignore. However, if pieces of the extender turn into accent marks, ornaments, etc., count them as misinterpreted symbol.

(12) ‘Notes’ are just that; treat rests and grace notes as ‘other symbols’, not as notes.

(13) Note with stem recognized as barline: count as both misinterpretation and missing note. Rationale: There’s certainly misinterpretation here, but missing a note is too important not to count it as such. Similarly, barline recognized as note with stem counts as both misinterpretation and extra note.

Appendix B. Guidelines for choosing OMR test pages (and factors affecting OMR accuracy)

This is a minor update of a 2008 MeTAMuSE/IU working paper (Guidelines4OMRTestPages.txt). A couple of new comments are in square brackets [ ], and ‘pages already scanned in suitable quality’ are now just ‘Preferred’ rather than ‘Strongly Preferred’. Otherwise, it simply removes a comment or two that turned out to be irrelevant, adds a reference to the current paper, and incorporates wording changes for clarity.

Factors affecting accuracy

We strongly suspect the factors below have a significant effect on the recognition accuracy of most, if not all, OMR programs; they should therefore be considered in choosing test pages.

(For MeTAMuSE tests, we generally wanted to minimize problems resulting from typographic style and image quality.)

(1) **CWMN complexity.** In general, more complex notation is likely to get worse results, as one would expect; but this topic is itself complex enough to deserve serious consideration. See the sections 'The challenge of variable complexity' and 'Complexity of the notation' in the current paper.

(2) **Typographic style.** 'Engraving' quality. Music of 'engraved' quality is likely to give the best results. Music of lower quality, but still with fixed-shape symbols (noteheads, rests, flags, accidentals, clefs, etc., as opposed to beams, slurs, etc.) having consistent shapes, is next best. Manually-produced ('manuscript' quality) images are very likely to give the worst results. (Some would say good results with them are hopeless unless the music is very simple.) Use of movable type or not. Music set from movable type – with, for example, staff lines that have numerous tiny breaks – may be considerably harder for programs to handle.
Conventions for shapes and positions of symbols. The standard 'modern' symbol shapes and positions are likely to give much better results. Some specific problem cases:

- Early 18th century editions, e.g. LeCene's of Vivaldi, with clefs significantly different from modern ones, sharps rotated 45 degrees, downstemmed half-noteheads on the wrong side of the stem, etc. These are not likely to work at all well, though it'd be interesting to try. (Is LeCene's style typical of early 18th-century editions? Probably so. Cf., e.g. R. Rastall, *The notation of western music*, p. 178.)
- The old Bach Gesellschaft editions, which use half-note heads without stems as whole notes. These probably won't be too hard for most programs to handle, and the Gesellschaft edition is too important to give up without a fight.
- Novello editions not set from movable type, many of which still incorporate odd things like backwards eighth rests (used for quarter rests) and backwards bass clefs (which function as normal bass clefs).
- Looseness of spacing. Music with very tight spacing, either horizontally or vertically (e.g. our Beethoven Trio test page), is likely to give worse results. (This is because OMR programs, like human readers, inevitably use proximity to disambiguate situations; of course programs can't do it in anything like an intelligent way.)

(3) **Image quality.** Print quality. Problems are likely to result from:

- Inferior quality reproduction, with breaks in stems, little holes in solid noteheads, etc.
- Poor contrast

[Byrd & Simonsen's 'Quality Levels' address the above aspects of print quality.]

- Bleedthrough (from the other side of the page)
- Markings written on the copy, typically by students and teachers

Geometry of image. Curvature, skewing, or other distortion of the image may lead to worse results. Curvature seems most likely to be a problem at the left and right edges, but the only occurrence in our collections we know of is at the *top* of p. 9 of Haydn Symphony no. 1. (In general, image geometry problems are more likely artifacts of the scanning process rather than features of the page scanned.)

Miscellaneous Factors

The following factors unrelated to accuracy should also be considered in choosing test pages. Strongly preferred (and the more factors apply, the better):

- public-domain editions
- public-domain music
- pages with high-quality, fully symbolic encodings also available (for use as ground truth)
- pages used in previous OMR studies, e.g. our own SFStudy, Bellini et al.'s seven pages, and the OMR Quick-Test; also the CCARH 1994 studies (especially a 1907 edition of Haydn's Symphony no. 1, but perhaps also a bunch of short examples) and Bainbridge's 'cantor' collection

[IMSLP has great potential as a source of pages fulfilling at least the first three of the above factors.]

Also preferable (though not strongly):

- pages already scanned in suitable quality
- fist pages of movements (to avoid problems with ill-defined context, especially the fact that most other pages don't have a time signature at the beginning of the page)
- pages with encodings available that aren't clearly high-quality, or encodings as MIDI files (but NB: a MIDI file encoding that tries to capture details of a performance is less likely to be useful than one based entirely on a score)

Appendix C. 'Pianoform' music notation

Pianoform music notation involves the following features:

- Structural interaction between the staves (visible in Figure 10 to a great extent, and Figures 8 and 11 to a lesser extent)
- Structural interaction within a staff: interaction leading to three or more horizontal positions for the same logical time (in Figures 7 and 10*); voices appearing and disappearing anywhere (Figures 6 and 12); individual noteheads belonging to two (Figures 7, 8, and 12) or even three voices; interlocking beams (Figure 12); 'impossible rhythms' (Figure 12**)

Note also two subtle but extraordinary details of our figures:

* Figure 10 is in 3/8 time, but the left hand in the 4th measure of the top system appears to have a total duration of six eighth notes. But its total duration is actually only the expected three eighths, because the first chord after the clef change is on the downbeat.

** In Figure 12, look at the last note in the right hand. With the upstem, it is a normal 16th, starting a 16th before the barline; with the downstem, it is a quintuplet 16th, starting a shorter time before the barline. Of course the same rhythm continues throughout this excerpt. Hook (2011) discusses what he calls 'impossible rhythms' in 19th century piano music; this is one of about 50 examples he cites.

While the five figures we cite are all from piano music, an obvious question is whether these phenomena are really characteristic of music for the piano and not other instruments. It would be hard to make the case that any of these features *never* occur in music for other instruments. But we have never seen most of them in music for anything other than piano; and, for the following reasons, it is plausible that they are far more common in music for piano than for anything else. (Note that we are considering only music for 'popular' instruments, including human voice.)

(a) The piano is widely used to play reductions of orchestral scores for accompaniments (often with large numbers of notes and voices in a very wide range)

(b) The piano has a sustain pedal (allowing the player to keep any number of notes sounding at once)

(c) It has a huge range: normally 88 semitones, the widest of any popular instrument (resulting in, e.g. voices needing to change clefs, sometimes by changing staves, and ottava signs)

(d) It has been very popular from the 19th century on (when music became more complex and composers more interested in specifying details than previously)

No other popular instrument has feature (a). The only other one that has (b) is electric guitar, but its music is almost always from a different musical tradition – rock music – with far less notational complexity. And the only other popular instruments that come close on (c) are the harp and the organ. All, or virtually all, harps have a range of about 81 semitones. Large pipe organs can have a range even wider than the piano, but an organ keyboard rarely has more than 61 semitones, and a pedal board far fewer – say, 32 at the most. An organ achieves a wider range with buttons to shift keyboards and pedals by octaves. Also, very low notes are played by the feet on the pedals, so there is limited interaction with higher notes. Finally, (d) is particularly relevant to distinguishing piano music from music for earlier keyboard instruments, especially the harpsichord.