Optical Music Recognition with Human-Labeled Constraints

Liang Chen
Indiana University, Bloomington
Bloomington, IN, USA
chen348@indiana.edu

Erik Stolterman
Indiana University, Bloomington
Bloomington, IN, USA
estolter@indiana.edu

Yucong Jiang
Indiana University, Bloomington
Bloomington, IN, USA
yucong.jiang@gmail.com

Christopher Raphael
Indiana University, Bloomington
Bloomington, IN, USA
craphael@indiana.edu

Rong Jin
Indiana University, Bloomington
Bloomington, IN, USA
rongjin.jr@gmail.com

CHI'14, April 26–May 1, 2014, Toronto, Canada.
Copyright © 2014 ACM ISBN/14/04...$15.00.
DOI string from ACM form confirmation

Abstract

We present our *Ceres* system for human-directed optical music recognition (OMR). Given the state and progress of OMR it seems unlikely that a fully automated system will be developed soon. For this reason we explore an interactive approach that allows the user to offer guidance through the labeling of individual pixels with particular symbols. The system takes these labels as constraints and produces interpretations consistent both with the human-supplied labeling and the underlying grammatical descriptions of the objects. The approach puts a good deal of flexibility and power into the user's hands, without requiring the user to understand the system's inner-workings. Furthermore, the ideas generalize to a broad range of human-directed recognition problems. We show several specific recognition examples in detail, and also give a video that presents a bird's-eye-view of the system behavior and user interaction.

Author Keywords

human-in-the-loop computing; optical music recognition; document recognition

ACM Classification Keywords

H.5.5. [Systems]

Introduction

Optical music recognition (OMR), the musical cousin of optical character recognition (OCR), seeks to convert music score images into symbolic representations. OMR has attracted somewhat sporadic interest from various communities for more than 40 years, though the current state of the art [7, 1], falls far short of what is needed to accomplish our goal: harvesting large scale symbolic music libraries [6, 5]. The problems are that OMR poses many difficult modeling challenges, lacks any obvious recognition paradigm, and is thwarted by a thicket of special cases and exceptions to general rules. As with many difficult recognition problems, correct interpretation requires explanations that "make sense," both in terms of local context as well as broader notational conventions. This sense is easy for humans to grasp but hard to distill into algorithmic approaches.

As it seems unlikely that any fully automated approach will soon be developed, we propose an interactive solution in which a person guides the system toward the desired end. The essential idea allows the user to correct recognition errors by labeling individual pixels according to a set of *primitive* symbols such as note head, stem, beam or accidental. The system then re-recognizes *subject to this constraint*. This interaction is iterated until the human-computer team converges on a correct interpretation.

This work is an example of a *Mixed-Initiative System* [3, 9, 2, 4], though the pattern of interaction in our system is relatively deterministic. The "communication issue," which deals with the way human and computer must convey their agendas and understanding to each other, figures prominently in the current effort as we must facilitate simple yet expressive human input. Such "human-in-the-loop" systems are currently gaining traction in the object recognition and vision communities [8, 11]. Here too the difficulty of efficiently modeling the channel between human and computer is a central challenge.

We see three principal virtues that argue for our approach:

Simplicity. Our method does not require the user to have any detailed knowledge of the inner-workings of the system. Rather, the user needs only to identify errors in the end result in terms of the *primitive* that should "cover" a particular pixel. For instance, "this pixel should be part of a sharp." Musical symbols are often composed of highly constrained configurations of primitives, such as beamed groups and chords, while these *composite* symbols are generally suited to grammatical representations, as described in the next section. When the user expresses a pixel constraint our system must find an image interpretation consistent both with the user input and the interrelated grammatical constraints that govern the recognition engine. However, the user need not be aware of this formulation. Rather she operates in the "Flatland" world of primitive coverings while the system operates in a higher-dimensional domain.

Power. Due to our grammatical formulations of musical symbols, a single constraint often calls into question a number of decisions made during recognition. Thus incorporating such constraints often resolves several mistakes at once. Furthermore, as the user only constrains the "Flatland" world, the grammatical representations that create *meaningful* primitive configurations are left intact. The human-generated constraints can be applied in many ways, giving the user a great deal of flexibility and variety in directing the outcome.

Generality. While our emphasis is on OMR there is nothing in our human-directed approach specific to this domain. We formulate the recognition problem as dynamic-programming-based optimization, which is a completely

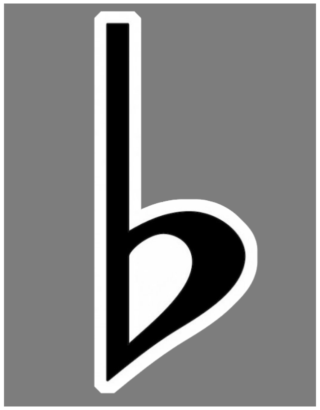

Figure 1: Our templates partition a rectangular region into pixels labeled as *black*, *white*, and *unknown*, corresponding to the p_B, p_W, p_U probability models

generic view leading to a broadly applicable human-in-the-loop recognition strategy without reference to any particular recognition domain or kind of data.

Fleshing Out our Idea

Our approach is phrased in terms of a simple data model that measures how well a particular hypothesis agrees with the pixel data grey levels, $g = g(x)$ where x is an image site. Figure 1 shows an example of a template for a flat. Our symbol templates partition a rectangle of pixels into 3 categories: *black* (B), *white* (W) and *unknown* (U), each corresponding to a grey-level probability distribution. Given a template and its location we assume the grey levels are independent:

$$p(g) = \prod_{x \in B} p_B(g(x)) \prod_{x \in W} p_W(g(x)) \prod_{x \in U} p_U(g(x))$$

Our actual scoring function normalizes every pixel factor above by the *unknown* model, estimated by the the grey-level distribution taken over the entire image, leading to:

$$H(B, W) = \sum_{x \in B} \log \frac{p_B(g(x))}{p_U(g(x))} + \sum_{x \in W} \log \frac{p_W(g(x))}{p_U(g(x))} \qquad (1)$$

One can see that the *unknown* pixels contribute nothing to the sum of Eqn. 1, thus we can view the equation as a model for the entire image where the majority of the pixels are treated as *unknown*. This allows us to make meaningful comparisons between hypotheses (B, W, U regions) of different size.

OMR presents a wide variety of different recognition problems such as identifying the staves in the page, decomposing the staves into systems and measures, as well are recognizing the contents of the various measures. We pose *all* of these recognition tasks as optimizations of Eqn. 1 over families of hypotheses that partition the image into *black*, *white*, and *unknown* regions. For instance, the simplest example would be identification of isolated symbols, accomplished by maximizing Eqn. 1 over possible translations of the template.

Most of music notation consists of *composite* symbols — configurations of *primitives* (note heads, stems, flags, accidentals, etc.) that fit together in grammatically constrained ways. Perhaps the simplest example would be the clef-key-signature group that typically appears at the left edge of every staff. We use this example to motivate our ideas. The clef-key-signature group is composed of one of (say) 3 possible clefs, and between 0 and 7 flats or sharps. The vertical position of the clef on the staff is fixed by convention, while the vertical position of each accidental is determined by the clef and its position in the sequence. For instance, with a treble clef the 2nd sharp would appear on the 2nd space from the top of the staff. The possible configurations of clefs and accidentals are summarized by the graph in the left panel of Figure 2; each legitimate configuration is associated with a path through the graph.

As is often the case, such a generative model can be turned into a recognition engine. Suppose we have a state sequence $s_1, s_2, \ldots, s_K$ corresponding to a path through the graph in Figure 2, as well as a *partition* of our interval of explanation, I, into subintervals, $I_1, I_2, \ldots, I_K$, as shown in the right panel Figure 2. The $\{s_k\}$,$\{I_k\}$ collectively define a presentation of the clef-key-signature in which the $\{s_k\}$ determine the symbols and their vertical positions, while the $\{I_k\}$ determine their horizontal positions. Each hypothesis, $s_1, \ldots, s_K$, $I_1, \ldots, I_K$ is scored by our normalized data model

$$H(B, W) = \sum_{k=1}^{K} H(B_k(s_k, I_k), W_k(s_k, I_k)) \qquad (2)$$

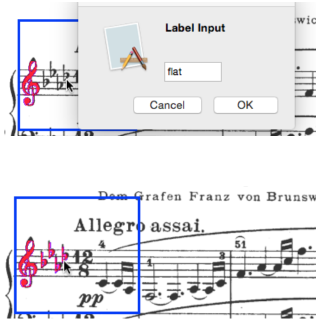

Figure 3: Top: Misrecognized clef-key-signature with a pixel labeled as *sharp*. **Bottom:** Optimizing the objective function subject to this constraint fixes these errors.

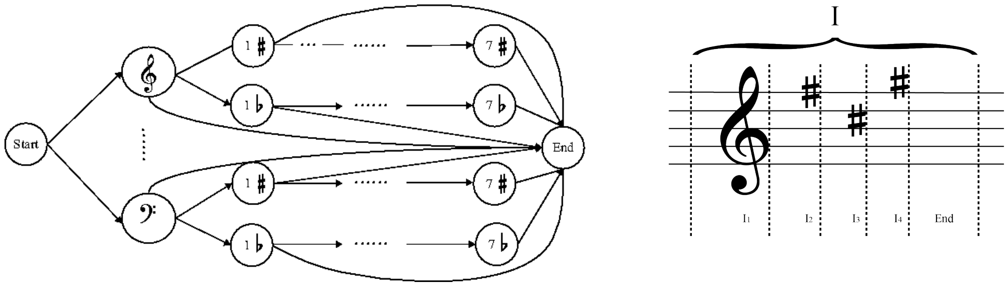

Figure 2: Left: The state graph showing the possible sequences of clef and accidentals composing a meaningful clef and key signature. Right: A labeled partition as produced by our dynamic programming approach.

which is easy to optimize using dynamic programming (DP). Such recognition strategies have been a mainstay of OCR for many years [10], with many variations on this idea common in computer vision and many other recognition domains. While simple, such grammatically-formulated approaches are surprisingly flexible. In addition to clef-key-signatures, our OMR system uses this approach to recognize chords, beamed groups, slurs, "hairpin" dynamics, text dynamics, staves, and systems.

This recognition approach accommodates human-supplied constraints naturally. Consider the top panel of Figure 3 in which we display a misrecognized result over the original image — here the recognizer identified the correct clef but missed the four flats that complete the clef-key-signature. While clef-key-signatures are comparatively easy to recognize, such errors can occur before our system has had a chance to adapt to the particular fonts in a document. In such a case our approach allows the user to label any pixel as a member of a particular symbol, while the system re-recognizes subject to this constraint. The top panel of the figure shows the user identifying a *flat* pixel in the 4th flat of the key signature. It is a simple matter to optimize Eqn. 2 subject to this constraint: we only need modify our objective function to give a score of $-\infty$ to any template hypothesis (s_k, I_k) that mislabels the identified pixel. The DP calculations proceed identically aside from this change, resulting in the solution given in the bottom panel of Figure 3.

Since the grammatical structure greatly limits the meaningful presentations of a clef-key-signature, a single constraint can often fix multiple aspects of a failed recognition attempt. Note that the constraint in our example does not specify *which* flat the pixel belongs to, however, along with grammatical constraints on symbol separation, it is enough to tip the balance in favor of the correct interpretation, thus resulting in the four correctly identified flats of the key signature.

We cast nearly all recognition in OMR as optimizations of our data model, Eqn. 1, subject to grammatically defined constraints. The ideas of this human-in-the-loop strategy apply to all recognition problems for our system. While the grammatical formulations of the recognition problems can be complex, the user does not need to understand these. Rather, her task is simply to label individual pixels in a way that brings about the desired result. The user can approach

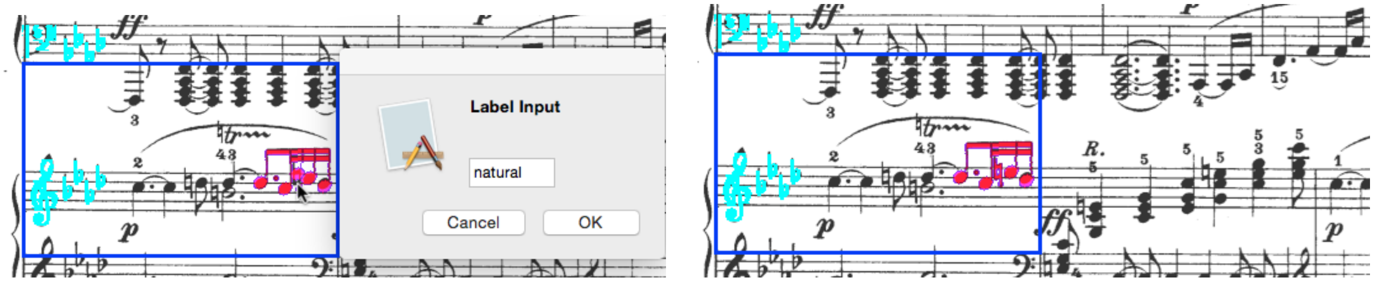

Figure 4: Top: A natural sign is misrecognized as a note head with accompanying stem. The use labels the indicated pixel as part of the natural. Bot: Using this constraint the system correctly recognizes both the primitives and their underlying grammatical structure.

the task naively, simply labeling misrecognized portions of the image, though experienced users find more efficient paths to the desired result.

The heart of our system is *measure recognition*, where the human-computer team recognizes the contents of each measure, one composite symbol at a time. For each measure the system identifies and presents possible candidates for recognition, including beamed groups, chords, slurs, hairpin dynamics, rests, etc. The user toggles through these, selecting one to recognize (or adding her own). If the candidate is recognized perfectly the user simply blesses the example and moves on. Otherwise the user and system iterate the process of labeling individual pixels and re-recognizing subject to all accumulated constraints. After a recognized symbol is accepted the corresponding region cannot be invaded by subsequent recognition processes.

As measures contain a great variety of symbols, there are quite a few possible labels the user can select for a pixel. At present these include various note heads, stems, flags, augmentation dots, accidentals, beams, rests, mid-line clefs, slurs, hairpins, fingerings, articulations and ornaments.

Figure 4 shows an example of our system in action. In the original recognition the program misidentifies the natural sign as an additional note head attached to the beam group. The user then identifies a pixel within the natural, labeling it accordingly, thus directing the system to re-recognize subject to this constraint. The grammatical representation of a beam group informs the system that, if it cannot put a note head where the natural is, it must also sacrifice the stem that connects to the beamed group. The result of re-recognition not only contains the correctly registered primitives, but also the grammatical relations from which they derive their meaning. For instance the resulting beam group representation understands the ownership associations between note heads, beams, accidentials, etc. that give rise to the figure's rhythm and pitch. Thus the user's single pixel labeling results in significantly more than substitution of an accidental at one image position for a note head and stem.

The video at http://music.informatics.indiana.edu/papers/hcml16/human_omr_actions.mp4 gives a bird's eye view of our in-progress *Ceres* system. The video contains a single frame for each user action, shown at 5 frames per second. Thus the 81-seconds account for 405 user actions. One can see the symbolic data accumulating in blue while the current object of the system's attention is shown in red, thus giving a crude sense of the level of human effort involved in the process.

Remaining Challenges

Our dream is that the eventual *Ceres* system will be widely adopted in a distributed citizen-music-scholar effort to create a large-scale, open, definitive, symbolic music library. For this to happen the system must produce high quality results quickly enough to justify its use: *user time* is the most relevant evaluation metric. While we cannot say exactly what the average time per page must be, we doubt we are at this point yet. One problem is that our recognizers make too many mistakes, forcing the user to coax the system through scenarios that could perhaps be recognized automatically. We continue to strive to improve our core recognition ability both through appropriate model specification and automatic learning.

References

[1] Baird H.S. Blostein D. 1992. A Critical Survey of Music Image Analysis. In *Structured Document Image Analysis*, H. Bunke H. S. Baird and K. Yamamoto (Eds.). Springer-Verlag, Berlin, 405–434.

[2] Veloso M. M. Cox M. T. 1997. Supporting Combined Human and Machine Planning: An Inteface for Planning by Analogical Reasoning. In *Proc. 2nd Int. Conf. on Case-Based Reasoning*, D. Leake and E. Plaza (Eds.). Springer, Berlin, 531–540.

[3] Allen G. Ferguson G. 1998. TRIPS: An Intelligent Integrated Problem-Solving Assistant. In *Proc. of the 15th National Conf. on Art. Intel.* Menlo Park, CA, 567–573.

[4] Myers K. L., Jarvis P. A., Tyson W. M., and Wolverton M. J. 2003. A Mixed-Initiative Framework for Rubust Plan Sketching. In *Proc. 13th Int. Conf. on Automated Planning and Scheduling*, Dana Nau Enrico Giunchiglia, Nicola Muscettola (Ed.). AAAI Press, 256–265.

[5] Jin R. Raphael C. 2014. Optical music recognition on the International Music Score Library Project. In *Document Recognition and Retrieval XXI*.

[6] Wang J. Raphael C. 2011. New Approaches to Optical Music Recognition. In *Proc. of the 12th Int. Conf. on Music Info. Retrieval*. ACM Press, Miami, USA, 305–310.

[7] Cardoso J. S. Rebelo A., Capela G. 2009. Optical Recognition of Music Symbols. *International Journal on Document Analysis and Recognition* 13 (2009), 19–31.

[8] Branson S, Wah C., Babenko B., Schroff F., Welinder P., Perona P., and Belongie S. 2010. Visual Recognition with Humans in the Loop. In *European Conference on Computer Vision (ECCV)*. Heraklion, Crete.

[9] Cox M. T., Edwin G., Balasubramanian K., and Elahi M. 2001. Multiagent Goal Transformation and Mixed-Initiative Plannng Using Prodigy/Agent. In *Proc. 4th Int. Multiconf. on Systemics, Cybernetics and Informatics*, Vol. 7. Orlando, FL, 1–6.

[10] Govindan V.K. and Shivaprasad A.P. 1990. Character recognition – A review. *Pattern Recognition* 23 (1990), 671–683.

[11] Alexander Waibel and Rainer Stiefelhagen (Eds.). 2009. *Computers in the Human Interaction Loop*. Springer Verlag.