CERES: AN INTERACTIVE OPTICAL MUSIC RECOGNITION SYSTEM

Liang Chen, Christopher Raphael
Indiana University Bloomington, USA

ABSTRACT

We present our Optical Music Recognition (OMR) system, *Ceres*, which embeds human intelligence into machine recognition. Different from previous OMR systems, which took human corrections as an extra post-processing step, our systems accepts user-supplied information during the recognition process. Due to the big challenge of OMR, all the existing systems suffer from different levels of errors. The inadequate recognition accuracy impeded the practical use of these systems, which always requests a large amount of human labor for error corrections. This proofreading process is time-consuming, laborious and sometimes error-prone. To alleviate the burden of proofreaders, we apply human-in-the-loop computation to encourage tight collaboration between human and computer, where each can contribute its best. In our system, user is only responsible for providing a small amount of information while the recognition and registration of symbols are left to the computer. We demonstrate our idea through an interactive OMR system, which shows the potential to considerably save the human efforts for OMR proofreading and be useful in practice.

1. INTRODUCTION

Optical Music Recognition (OMR) holds potential to transform score images into symbolic music libraries. The development of OMR technics, however, has been disappointingly slow. Even the best systems, which appear to be commercial, leave much to be desired [2]. The reason for these disappointing results is simply that OMR is *challenging* [1]. Rebelo [4] suggested that an interactive OMR system could be a realistic solution to the problem. *Ceres* uses human feedback as a means of constraining the recognition process, thus leveraging the user's input in the heart of the system.

We pose the problem as one of the *constrained* optimization. The first type of constraint comes from the grammar that governs the generation of music symbols. The other type is the user-imposed constraint. For instance, user can click on one pixel and label it as part of one specific *primitive* (the smallest unit that composes music symbols). There are around 40 different primitive labels offered by our system, which cover different *note heads*, *flags*, *beams*, *stem*, *accidentals*, *dynamics*, *clefs*, *rests*, *slur*, etc. User can also provide another two region-based constraints: "reuse a region" (after the region was accounted for by previously recognized symbols), and "white space" (there is no primitive to be recognized in this region). The recognition is, therefore, driven by both the built-in and dynamic constraints.

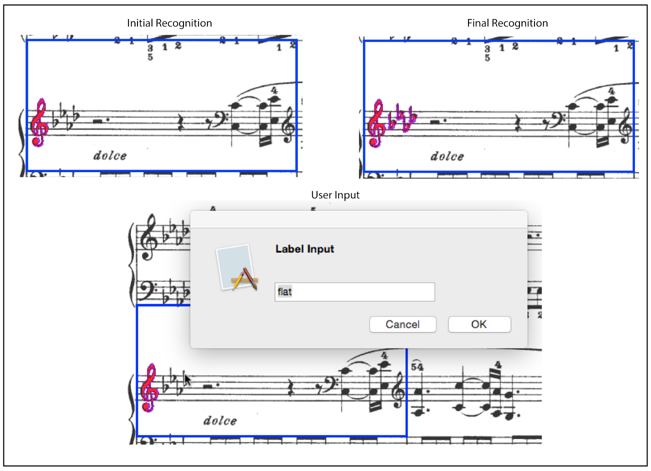

Figure 1. The system interface, recognition process and re-recognition with user-supplied constraint

Ceres consists of three key modules: *staff finding*, *system identification* and *symbol recognition*. The first two decode the whole image into separate measures, while the last one recognizes music symbols in each measure. The system accepts user-supplied constraints during each of these three steps, which will be delivered to the recognizers for a second recognition. For instance, if the staff finder missed one staff, the user could click on any pixel of the missing staff and input "staff line" (positive constraint) to enforce a second staff recognition subject to the given constraint. Another case is when the system recognized an extra bar line, the user might point out the incorrectly recognized region and tell the recognizer it's "white space" (negative constraint); then the system would come up with another interpretation without violating this constraint.

We designed an easy-to-use interface to take these user-supplied constraints (including locations and labels) and reflect the re-recognition results automatically (Fig. 1).

Attribution: Liang Chen, Christopher Raphael. "Ceres: An Interactive Optical Music Recognition System", Extended abstracts for the Late-Breaking Demo Session of the 16th International Society for Music Information Retrieval Conference, 2015.

2. THE MODEL

For all recognition components of our system, including staff finding, system identification, and symbol recognition, we formulate the essential tasks as optimization problems. Letting x denote a pixel location in the image, and $I(x)$ the grey level intensity at x, we have four types of probability models for these intensities indexed by $\mathcal{M} = \{b, w, t, n\}$. These correspond to pixels we believe to be *black*, *white*, *transitional*, and *null* with the probability models denoted by p_b, p_w, p_t, p_n. For a possible image interpretation, H, we assign each image pixel to one of the four models through the function $M_H(x)$. We compute the score of a particular hypothesis as

$$S_H = \sum_x \log \frac{p_{M_H(x)}(I(x))}{p_n(I(x))} \tag{1}$$

In theory, the sum extends over the entire image, though hypotheses generally label many pixels as *null*, in which case they only contributes 0's to the sum of Eqn. 1. Our approach for all phases of recognition begins by optimizing S_H subject to the inherent grammatical constraints on the hypothesis [3]. In formulating human-directed recognition, we allow the user to introduce various constraints by labeling individual pixels. Thus, at any point in our interactive computation we have a collection of user-supplied constraints, $C = \{(x_i, l_i)\}$ for $i = 1, \ldots, n$, meaning that the user forces the pixel at x_i to be labeled as l_i. From these constraints we develop an additional term to our objective function

$$T_H = \sum_x t(x, P_H(x))$$

where $P_H(x)$ is the label of location x according to the hypothesis, H, and

$$t(x, P_H(x)) = \begin{cases} C & x = x_i, P_H(x) = l_i \text{ some } i \\ -C & x = x_i, P_H(x) \neq l_i \text{ some } i \\ 0 & \text{otherwise} \end{cases} \tag{2}$$

Thus the objective function gives a *bonus* of C whenever the user-specified constraint is satisfied, and a *penalty* of $-C$ if it is not. While the negative constraints (uses of $-C$) are redundant in theory, they allow us to compute hypothesis scores by summing Eqn. 1 only over the relevant (not *null*) pixels. Our constrained objective function is then

$$Q_H = S_H + T_H. \tag{3}$$

We choose C large enough so that only hypotheses respecting the constraints can be found by the recognition engine. The technical details of *symbol recognition* process will be elaborated in Section 3. Illustrative examples of human-directed *staff finding* and *system identification* can be found through the link below [1] .

3. SYMBOL RECOGNITION MODULE

The core part of the system allows the user to participate in the *symbol recognition* in an interactive and incremental (symbol by symbol) way. We choose to work on the symbol level in consideration of the speed of system response. When the user clicks on one measure, the system will highlight the working region of that measure with a bounding box, and automatically detect the candidates in four different categories: *beamed groups*, *isolated notes*, *slurs* and *hairpin crescendoes*. The candidates will be ranked based on the candidate scoring function. User can toggle through all these candidates to choose the appropriate one to continue working on. Or they can choose to recognize all other rigid symbols (such as rests and small clefs) in a batch. After choosing the candidate, the user can type 'r' to recognize the target symbol. If there is any error on the recognized symbol, the user may want to label one pixel or a certain region to make the correction. After the symbol is fully corrected, the user can type 's' to save it. During the saving process, the system will automatically train the newly-identified primitives. This will gradually improve the performance of the recognition, and thus further increasing the efficiency.

4. CONCLUSION

We have built a prototype system that takes the score images as input, decode the page and recognize the musical symbols with the participation of human decisions. Different from other former systems, which treated human corrections as a separate post-processing step, we fuse the human intelligence organically into the recognition. The symbol recognition is highly grammatically constrained, so a small amount of user-supplied information may yield huge improvement in recognition. We kept making progress on our human-driven OMR system so that it could be more intelligent, robust and efficient. We figure this is a valuable opportunity for us to present *Ceres* in front of the ISMIR community. We also look forward to any suggestions that help us improve the system during the Late-Breaking session.

5. REFERENCES

[1] David Bainbridge and Tim Bell. The challenge of optical music recognition. *Computers and the Humanities*, 35(2):95–121, 2001.

[2] Donald Byrd and Megan Schindele. Prospects for improving omr with multiple recognizers. In *ISMIR*, pages 41–46, 2006.

[3] Christopher Raphael and Jingya Wang. New approaches to optical music recognition. In *ISMIR*, pages 305–310, 2011.

[4] Ana Rebelo, Ichiro Fujinaga, Filipe Paszkiewicz, Andre RS Marcal, Carlos Guedes, and Jaime S Cardoso. Optical music recognition: state-of-the-art and open issues. *International Journal of Multimedia Information Retrieval*, 1(3):173–190, 2012.

[1] www.googledrive.com/host/0B5VHUijvJ3tmSUE1OWVVYVoyYjg