Optical Music Recognition of Early Typographic Prints using Hidden Markov Models

Laurent Pugin
Music Technology Area, Schulich School of Music
McGill University, Montreal, Canada
laurent@music.mcgill.ca

Abstract

Music printed with movable type (typographic music) from the 16th and 17th centuries contains specific graphic features. In this paper, we present a technique and associated experiments for performing optical music recognition on such music prints using Hidden Markov Models (HMM). Our original approach avoids the difficult and unreliable removal of staff lines usually required before processing. The modeling of symbols on the staff is based on low-level simple features. We show that, using our technique, these features are robust enough to obtain good recognition rates even with poor quality images scanned from microfilm of originals. The music content retrieved by the optical recognition process can be put to significant use in, for example, the creation of searchable digital music libraries.

Keywords: OMR, Early typographic prints, HMM

1. Introduction

Typographic scores from the 16th and 17th centuries are the most common music prints of that time [1]. Having an optical recognition system for this kind of document is therefore important, for example to build searchable digital libraries [2] or to assist musicologists in the realization of critical music editions [3]. But optical recognition of these documents is difficult because they are old and often in poor condition. Furthermore, they cannot be treated with conventional optical music recognition (OMR) techniques because of the numerous printing irregularities they present and their particular layout. One important point is that staff lines are not always continuous and their width can vary (figure 1).

The following section looks at the background of previous work carried out both in OMR of similar prints and in using HMM for OMR. In section 3 we explain how the research was organized and how our solution works. In section 4, we present the results we obtained on different data sets and with different parameters. Finally, section 5 discusses the results and further possible developments.

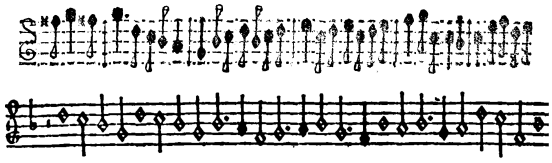


Figure 1. Example of staves printed with movable type

2. Background

2.1. Optical recognition of early typographic prints

OMR is a research field that has been covered by several studies [4]. Usually it is common music notation (CMN) that is treated, with solutions specific to its characteristics [5]. Optical recognition of early typographic prints has never been the subject of a specific study. Only Carter in the nineties once mentioned early typographic prints in his research on OMR [6]. The solution proposed by Bainbridge and Bell [7] is designed to be extensible, but the particularities of typography are not considered and never mentioned. The research by Pinto et al. [8] concerns more specifically early music but in manuscript form. The closest research is undoubtedly by Dalitz and Karsten [9] on recognition of prints of the same period but in lute tablature, proposing a solution specific to this music notation.

2.2. Hidden Markov Models in OMR

Hidden Markov Models (HMM) have almost never been used in OMR except for experiments by Kopec and Chou [10]. Our work was partly inspired by their use in optical character recognition, and more particularly in cursive script recognition. Two approaches are used to extract features from images, one segmentation-based and the other segmentation-free [11]. In the first, a segmentation algorithm is used to separate letters or letter elements. In the second, the system works exactly like most speech recognition systems based on HMM with a sliding window.

3. Experiments

Our approach deals with each staff globally without segmenting the symbols and exploits the fact that a staff is formed by a juxtaposition of typographic types from left to right. Our solution is model discriminant and uses the

typeface as the recognition unit. The idea is to build a direct correspondence between the music font used in the print and the set of HMM used for recognition. In consequence, the number of HMM, *i.e.* the system vocabulary, corresponds to the number of typefaces in the music font. Therefore the size of this vocabulary is about 200 different symbols. In the end, the HMM set will constitute a typographical model of a specific music font.

3.1. Training data

We built our training data from three different music prints [12, 13, 14] realized with two music fonts made up of very different graphical forms. To obtain a ground-truth of each staff, the musical content was entered via a MIDI keyboard in a music notation application. In all, 240 pages were entered, giving 1,478 staves and 52,178 characters corresponding to 175 different symbols.

We also developed a pre-treatment system for scanned pages, in which we use a scanning resolution of 400dpi with grayscale images. The image automatically extracted for each music stave is resized so that the distance between the bottom and the top staff lines is 100 pixels [15]. As well as saving time, the main advantage of this automatic pre-treatment solution (skew correction, size normalization, detection of staves on the pages, filtering, etc.) is that it allows our approach to be evaluated on original data and not on data prepared by hand.

3.2. Feature extraction

We use a segmentation-free approach and the feature extraction is performed with a sliding window as in speech recognition. We did not consider the segmentation-based approach because one of our goals was precisely to avoid segmentation problems. The features are extracted directly from the image without erasing the staff lines first. Our solution therefore treats the staff lines implicitly in the feature extraction. It is important to note that this moves away from other studies in OMR where removal of staff lines is a ubiquitous and complicated operation performed before the recognition task itself [16]. With our approach we can avoid this task, which would have been much more complicated with the prints considered here because of the irregularities of the staff lines.

We tried different experiments with various numbers of features – from 4 to over 40. Finally, we selected a solution with only 6 values chosen judiciously based on the staff configuration. Our tests showed their strongly discriminative nature and in any case additional values did not improve results significantly.

The values are calculated as follows: for each window, we determine the n distinct connected black zones with h and w the height and the width of the window, S its area and A the total area of the black pixels. The first value depends on the number of zones and corresponds to $\frac{1}{1+n}$ (this value is 1 if the window is empty). The second and third values are functions of the gravity centers c_x and c_y. The gravity centers are calculated according to the equations (1) and (2) where c_x^i corresponds to the gravity center x (respectively y) of a connected zone and a_i to its area. The used values are $\frac{1}{c_x}$ and $\frac{1}{c_y}$ and 0.5 for both if the window is empty.

$$c_x = \frac{\sum_{i=1}^{n} c_x^i \cdot a_i}{A \cdot w} \tag{1}$$

$$c_y = \frac{\sum_{i=1}^{n} c_y^i \cdot a_i}{A \cdot h} \tag{2}$$

The fourth feature corresponds to $\frac{a(n_i)}{S}$ where $a(n_i)$ is the area of the largest black element. The fifth value corresponds to $\frac{a(n_j)}{S}$ where $a(n_j)$ is the area of the smallest white element. If the window is empty, these values are 0 and 1 respectively.

The sixth value depends on the total area of the black elements in the window. This value is calculated with a weighting mask. The idea is to give more importance to the pixels that are between the staff lines than to those on the lines. The mask is designed in such a way that the nearer the pixels are to the center of a line space, the higher the weight factor is in the mask. In one sense, the window is stretched vertically, but the stretching is more important nearer the center of the line space than the line.

Finally, if I is the window and M the weighting mask, A', the total area of the elements after weighting, is given by equation (3) and the value used is given by equation (4). The value is 0 if the window is empty.

$$A' = \sum_{k=1}^{w} \sum_{l=1}^{h} I(k,l) \cdot M(k,l) \tag{3}$$

$$Feature = \frac{A'}{\sum_{k=1}^{w} \sum_{l=1}^{h} M(k,l)} \tag{4}$$

3.3. Implicit treatment of staff curvature

In some cases, staves can be curved on the image, for example if the page was not flat when the image was acquired. We use a two-stage process to consider this curvature in the feature extraction phase. The first stage enables the detection of the exact staff height. It is performed with a sliding window of 90 pixels. For each window, we make a horizontal projection from which we can deduce the staff height at that position. With this sliding window, the staff height detection performs well even if the staff is curved or leaning. For each staff we keep the median value, and for each page, the median of the values obtained for each staff.

In the second operation, we detect locally the vertical position of the staff. The principle is again to use a sliding window and to find for each window the staff position by correlating the horizontal projection of the window with a correlation mask representing the staff lines. The mask is built on the staff height detected (1 for the staff line and 0

elsewhere) with a line width of 5 pixels. A maximum of 20 pixels above and below a reference position is considered. The curvature detected is used in the feature extraction, for example when applying the weighting mask (figure 2).

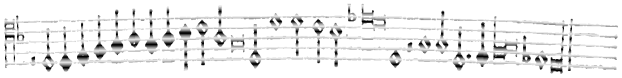
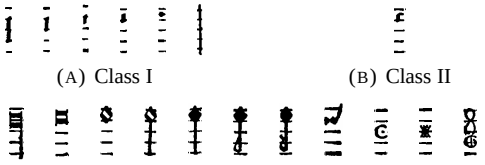


Figure 2. Staff curvature in feature extraction

3.4. HMM topology

We use left-right HMM because they model the sequential and unidirectional nature of the sequence we treat here well. Concerning the number of states, different experiments we made showed that the best results are obtained if the number of states of each HMM matches as closely as possible the width in pixels of the corresponding symbol. This characteristic has already been noticed by studies in handwriting recognition [17].

On the basis of the types found in the fonts of our training data, we defined three classes of topologies based on the type widths (figure 3). All types of our fonts fit these classes, but it is clear that with a font with types of another width (with oblique ligatures for example) it would be necessary to modify the number of classes.



(A) Class I (B) Class II

(C) Class III

Figure 3. Types of different topological classes

For one font, with w the size of the window and r the number of overlapping pixels each time the window is moved, the number of states of an HMM for each class S_c is defined by equation (5) where $P(c)$ is the width in pixels of the types of this class for this specific font.

$$S_c = \frac{P(c)}{w - r} \tag{5}$$

3.5. Space between symbols

Space between symbols can vary from one print to another but also within the same print or the same staff. Sometimes symbols are juxtaposed or spaced out by one or more spacing types (figure 4). To manage these different situations, we use a special model used for silences in speech recognition [18]. This model is considered neither in ground-truth sequences nor in recognition output.

Figure 4. Space between symbols

3.6. Distribution, initialization and learning

We used HTK [19] for our experiments with a continuous approach where for each state in the HMM output probability distribution is given by 5 gaussians. Usually, better results are obtained if only one gaussian is considered at the initialization and the number of gaussians is increased later [17]. Initialization of the models is done without ground-truth specifying the position of each symbol in the image. The training is then performed with the embedded version of the Baum-Welch algorithm. For every training iteration, each staff is used once to adapt the models corresponding to the symbols of which the staff is made.

4. Results

4.1. Evaluation sets

To evaluate our results, we used a set F_{MIX} which contains data regardless of the font used in the print, and two sets based on one print [13] to compare results obtained on the same data but in one case extracted from a cleaned facsimile edition (W_{FS}) and in another from a microfilm of the original (W_{MF}). This comparison enabled us to evaluate how robust our solution is to noise.

Table 1. Evaluation sets

Data	Staves	Characters	Symbols
F_{MIX}	1,478	52,170	175
W_{FS} and W_{MF}	491	22,290	152

Evaluation of the results is based on the edit distance between the recognized sequence of symbols and the correct one. Each time, we calculated the effective recognition rate (REC) and what we call the musical recognition rate (MUS) which does not consider height shift errors on symbols for which this has no musical consequence. For example, if a rest is correctly recognized but one line below or above, this is a recognition error which has no consequence from a musical point of view. For each evaluation, we performed a k-fold cross-validation with k=10. Because the ideal number of training iterations cannot be determined theoretically, we used the mean value of the results obtained from the final five training iteration models. We used 2 as window width with no overlap.

If we compare results obtained on W_{FS} and W_{MF}, we can see that the difference is small with less than 1.5% of loss with the microfilm images with most noise.

Table 2. Recognition rates

	F_{MIX}	W_{FS}	W_{MF}
REC	96.82	97.16	95.77
MUS	97.11	97.42	96.22

5. Conclusion

Whereas traditional OMR acts at symbol level using complex pattern recognition processes, we advocate the use of HMM for OMR of early typographic prints to operate directly at staff level. We show that by using well-chosen features and mono-dimensional models, accurate recognition can be achieved directly from the original microfilm without the need for unreliable pre-processing (*e.g.* staff-line removal). In particular, our recognition process integrates the staff curvature as well as the staff lines without any segmentation of the individual elements. We also show that our solution performs well even with image noise, which is essential when dealing with old documents.

With the topological classes proposed, the learning process is initiated without the need to specify the position of the symbols in the ground-truth, which greatly simplifies the creation of training data.

One innovative aspect of our recognition technique is that the staves are considered as sequences of symbols. This means that we can imagine building models of these sequences that can be integrated into the recognition process, enabling us to evaluate the validity or probability of a given musical sequence. The model could be a set of grammatical rules, *e.g.* for counterpoint, or a stochastic model, such as a class-based *n*-gram model adapted to music. This innovative approach to early music recognition opens up a whole range of new possibilities for future research.

The next stage should be to work on much more data in order to better evaluate whether it is more efficient to separate typographical models for each typeface or to use a single model for one or more typefaces.

Acknowledgments

This work was supported by the Swiss National Science Foundation through a research project at Geneva University, Switzerland, under the supervision of Prof. E. Darbellay.

References

[1] H. E. Poole, "Music Printing," in *Music Printing and Publishing*, D. W. Krummel and S. Sadie, Eds. New York: Norton, 1990, pp. 3–78, part one.

[2] G. S. Choudhury, T. DiLauro, M. Droettboom, I. Fujinaga, B. Harrington, and K. MacMillan, "Optical Music Recognition System within a Large-Scale Digitization Project," in *Proceeding of the 1st International Conference on Music Information Retrieval (ISMIR'00)*, Plymouth Massachusetts, 2000, online presentation.

[3] L. Pugin, "Computer Software for Early Music Editions: A New Approach," in *Music, Poetry, and Patronage in Late Renaissance Italy: Luca Marenzio and the Madrigal. International Conference, Harvard University, April 7-8 2006*, to appear.

[4] D. Blostein and H. S. Baird, "A Critical Survey of Music Image Analysis," in *Structured Document Image Analysis*, H. Bunke and K. Yamakato, Eds. Springer, Berlin, 1992.

[5] D. Bainbridge and T. Bell, "The Challenge of Optical Music Recognition," in *Computer and the Humanities*, vol. 35, 2001, pp. 95–121.

[6] N. Carter, "Automatic Recognition and Related Topics: Report for Research at Guildford, University of Surrey," *Computing in Musicology: A Directory of Research*, vol. 7, 1991.

[7] D. Bainbridge and T. Bell, "A Music Notation Construction Engine for Optical Music Recognition," *Software – Practice and Experience*, vol. 33, no. 2, pp. 173–200, 2003.

[8] J. C. Pinto, P. Vieira, and J. M. Sousa, "A New Graphic-Like Classification Method Applied to Ancient Handwritten Musical Symbols," *International Journal on Document Analysis and Recognition (IJDAR)*, vol. 6, no. 1, pp. 10–22, 2003.

[9] C. Dalitz and T. Karsten, "Using the Gamera Framework for Building a Lute Tablature Recognition System," in *Proceeding of the 6th International Conference on Music Information Retrieval (ISMIR'05)*, London, 2005, pp. 478–81.

[10] G. E. Kopec and P. A. Chou, "Markov Source Model for Printed Music Decoding," *Journal of Electronic Imaging*, vol. 5, no. 1, 1996.

[11] T. Steinherz, E. Rivlin, and N. Intrator, "Offline Cursive Script Word Recognition - a Survey," *International Journal on Document Analysis and Recognition (IJDAR)*, vol. 2, no. 2-3, pp. 90–110, 1999.

[12] E. Ghibel, *Il primo libro de madrigali a tre voci a note negre*. Venezia: A. Gardano, 1552, RISM G-1773, facsimile (Peer: Alamire, 1984).

[13] A. Willaert, *Fantasie recercari contrapunti a tre voci*. Venezia: A. Gardano, 1559, RISM W-1121, D-Mbs, microfilm and facsimile (Peer: Alamire, 1986).

[14] L. Marenzio, *Il quinto libro de madrigali a sei voci*. Venezia: A. Gardano, 1595, RISM M-0516, I-Bc, microfilm.

[15] L. Pugin, "Lecture et traitement informatique de typographies musicales anciennes. Un logiciel de reconnaissance de partitions par modèles de Markov cachés," Ph.D. Dissertation, University of Geneva, 2006.

[16] I. Fujinaga, "Staff Detection and Removal," in *Visual Perception of Music Notation: On-Line and Off-Line Recognition*, S. E. George, Ed. Hershey: IRM Press, 2005, ch. 1.

[17] S. Günter and H. Bunke, "Optimizing the Number of States, Training Iterations and Gaussians in an HMM-based Handwritten Word Recognizer," in *Proceedings of the 7th International Conference on Document Analysis and Recognition (ICDAR'03)*, vol. 1, Edinburgh, 2003, pp. 472–6.

[18] S. Young *et al.*, *The HTK Book (for HTK Version 3.2)*, December 2002, PDF version, <http://htk.eng.cam.ac.uk>.

[19] Cambridge University Engineering Department. Hidden Markov Model Toolkit. <http://htk.eng.cam.ac.uk>.