GAMERA *versus* ARUSPIX TWO OPTICAL MUSIC RECOGNITION APPROACHES

Laurent Pugin Jason Hockman John Ashley Burgoyne Ichiro Fujinaga
Centre for Interdisciplinary Research in Music and Media Technology
Schulich School of Music of McGill University
Montréal (Québec) Canada
{laurent,hockman,ashley,ich}@music.mcgill.ca

ABSTRACT

Optical music recognition (OMR) applications are predominantly designed for common music notation and as such, are inherently incapable of adapting to specialized notation forms within early music. Two OMR systems, namely Gamut (a Gamera application) and Aruspix, have been proposed for early music. In this paper, we present a novel comparison of the two systems, which use markedly different approaches to solve the same problem, and pay close attention to the performance and learning rates of both applications. In order to obtain a complete comparison of Gamut and Aruspix, we evaluated the core recognition systems and the pitch determination processes separately. With our experiments, we were able to highlight the advantages of both approaches as well as causes of problems and possibilities for future improvements.

1 INTRODUCTION

Optical music recognition (OMR) enables document images to be encoded into digital symbolic music representations. The encoded music can then be used and processed within a wide range of applications, such as music analysis, editing and music information retrieval. Over the years, multiple approaches have addressed this difficult task, in some cases focusing exclusively on one type of music document, such as keyboard music, orchestral scores, or music manuscripts [1]. Most commercial tools today are non-adaptive, acting as black-boxes that do not improve their performance through usage: when a symbol is misread, it continues to be misread in the subsequent pages, even if the user corrects the results by hand on every page. Attempts have been made to remedy this situation by merging multiple OMR systems, yet this remains a challenge [4].

Two research projects have taken an innovative approach to OMR by adopting adaptive strategies, namely Gamut, built based on the Gamera framework, and Aruspix. The main idea behind the adaptive approach in OMR is to enable the tools to improve themselves through usage by taking benefits from the training data that becomes available as the user corrects recognition errors. Adaptive approaches have been proven to be very promising with historical documents, because the very high variability within the data requires the tools to constantly adapt and retarget themselves. This is particularly true for early music prints from the sixteenth and seventeenth centuries, for which Aruspix was designed, as they contain an unpredictable variability of font shape, page noise, brightness and contrast [14].

This study is a first attempt at a comparison between Gamut and Aruspix. We specifically address the accuracy of the tools and their capability to adapt themselves to a new dataset. This paper is structured as follows. In section 2 we present a brief overview of the two OMR applications in comparison. In section 3 we present the experiments undertaken to enable such a comparison at different stages of the recognition process. Results are presented in section 4, and conclusions and future work are discussed in section 5.

2 INFRASTRUCTURE

2.1 Gamera and Gamut

Gamera is an open-source framework for the creation of structured document analysis applications by domain experts [8]. This system is designed for use with any type of image documents, and provides tools for preprocessing, segmentation, and classification of symbols within a document. The environment, structured as a composite of C++ and Python, provides facilities for developers to integrate plugins or toolkits suited for the specific type of documents being analyzed. Its framework is accessible enough to provide the tools for the development of an application for a specific domain without requiring a strong programming background. Within a Gamera application, individual symbols may be extracted by connected-component (CC) analysis and recognized using the *k-nearest neighbour* (kNN) classifier. If the symbols themselves are split during CC analysis, the kNN classifier is capable of automatic regrouping and recognition, which has been shown to be a necessary feature when working with degraded documents [7].

Gamut is an OMR application built with the Gamera framework [10]. It is comprised of several plugins and toolkits that provide procedures specific to the task. As in most OMR systems, Gamut requires additional preprocessing to remove staff lines prior to symbol recognition, a task that is especially important for subsequent pitch detection from classified symbols [5]. Within Gamut, staff detection and removal is performed by the MusicStaves toolkit, a collection staff-removal algorithms specialized for specific types of musical documents.

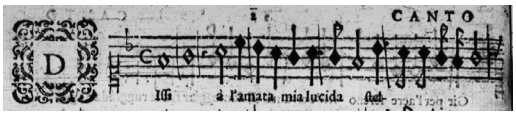
2.2 Aruspix

Aruspix is a cross-platform software program for OMR of early music prints. Aruspix performs the entire OMR process, from preprocessing to score encoding. The preprocessing stage includes document deskewing, border removal, binarization, and a heuristic approach to preclassify each region of the image as text, music, or an ornate letter. Once preprocessing is complete, Aruspix reads the musical content of each page and converts it into an editable digital music format. Early music has been a challenge for traditional approaches to OMR [11], and so Aruspix has taken a unique approach based on hidden Markov models (HMMs) [12, 13]. The two key features of the method are that it does not require staff removal and that pitch detection is integrated within the recognition process. Aruspix also provides a built-in music editor to assist in correcting recognition mistakes. Features like this one make Aruspix usable not only as a research tool but also as an end-user application.

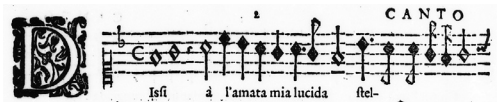
3 EXPERIMENTS

To perform our comparison, we evaluated both systems on the same set of pages to measure their accuracy. In particular, we were interested in comparing individual symbol recognition rates in addition to the overall performance of both applications. For these experiments, we used a data set built from four books of madrigals for four voices, composed by Luca Marenzio (1533–99). The books were printed in Venice and Antwerp between 1587 and 1607 (RISM [15] M-0579, M-0580, M-0583 and M-0585). As they are all re-editions of the same collection of madrigals, they contain the same musical content, with very few exceptions. The uniformity of content ensures that performance variations across books within our experiments are mainly related to the graphical appearance of the music fonts and document degradation, rather than the musical content itself.

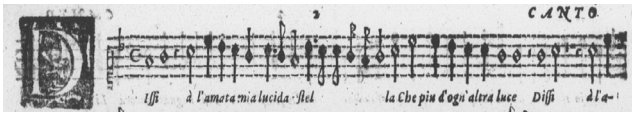
From each book, we selected 40 pages for the training sets and 30 pages for the testing sets, resulting in total of 280 pages. For the training sets, we chose the first 20 pages of both the *Canto* and the *Alto* parts. This was not an arbitrary decision, as it is in this order that the pages would logically appear if either Gamut or Aruspix were used in a real-life project, as one would most likely start from the beginning of the books and continue onward. For this reason, and as the intent of the experiment was to reflect a comparison of tools rather than their absolute performance, we opted not to cross-validate our results. Further, it has been shown that cross-validated results do not vary significantly in such experiments [14].



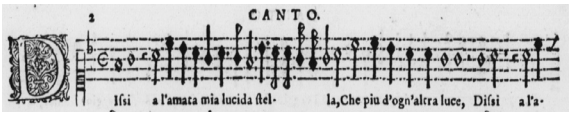
(a) RISM M-0579 (R. Amadino, Venice, 1587)



(b) RISM M-0580 (G. Vincenti, Venice, 1587)



(c) RISM M-0583 (A. Gardano, Venice, 1603)



(d) RISM M-0585 (P. Phalèse, Antwerp, 1607)

Figure 1: Prints used for the experiments

Within Aruspix, each image underwent preprocessing with skew-removal and binarization, and all borders and non-musical symbols (e.g., titles, lyrics, and ornate letters) were removed. Because some image features used for classification are size dependent, the image size was then normalized to a staff height of 100 pixels. Ground-truth data were generated by transcribing the entire dataset using the Aruspix integrated editor.

3.1 Preprocessing in Gamut: staff removal

Staff-line removal is known to be very difficult within early documents due to numerous printing irregularities and frequent document degradations [9]. Dalitz et al. have recently presented an extensive comparison of several staff removal algorithms using self-generated degraded images and three different error metrics to gauge their performance [5]. This study clearly demonstrates that of the six algorithms tested, no single algorithm outperforms the others across all the data. As this evaluation does not focus on early music documents in particular, it was not possible to assume which algorithm would be most suitable for our printed sources.

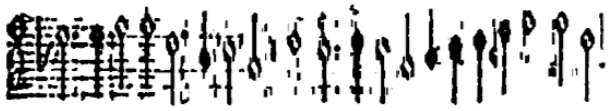


Figure 2: Typical staff removal error in M-0583

Moreover, as the study was based on artificial data, we could not be certain that a best candidate would be ideal for *real* data. For this reason, a preliminary experiment was undertaken to determine the most suitable algorithm for our purposes within the MusicStaves toolkit. A test database was created comprised of 70 hand-selected images chosen to provide a reasonable sampling of possible pages that one would encounter within this historical period.

For our purposes, it was easy to determine the three best algorithms amongst those tested (*simple*, *carter*, *roach-tatem*, *linetracking*, *fujinaga*, and *skeleton*) through a subjective evaluation of the results, as some of them failed dramatically. We have subdivided the errors accumulated over the data into minor and major errors. Minor errors are determined to be local to an individual or pair of symbols; effects to symbols beyond those affected were negligible. Alternatively, major errors encompass more destructive errors, usually including several symbols [1] . *Carter* was invariably unable to detect most staves, while *simple* and *roach-tatem* proved too destructive to symbols. These methods were significantly outperformed by the *linetracking*, *fujinaga*, and *skeleton* methods. In most images, these three algorithms were capable of the identification and removal of a majority of staves, including cases in which the others algorithms encountered major problems. We found the *linetracking* method to be most effective while avoiding major symbol deterioration, yet on occasion, noticeable problems were still encountered (figure 2).

3.2 Training and optimization

With all the data in hand, we were able to train HMMs for Aruspix and kNN classifiers for Gamut towards a comparison of the two systems. To measure the learning rates of both systems, we trained different HMMs and kNN classifiers using different numbers of pages of our training sets, from 1 to 40. Each model or classifier was then evaluated against the same testing set, such that we were able to determine the change in both systems once trained on 1 to 40 pages. We chose not to mix pages from the different books as we wanted the models and classifiers to be highly book specific in order to evaluate the graphical characteristics that make either system perform better or worse.

[1] A comprehensive discussion of the results of this evaluation, as well as supplementary images are available online at `coltrane.music.mcgill.ca/DDMAL/`

Both Gamut and Aruspix core recognition systems can be optimized with specific approaches that have been evaluated as well. The performance of the Gamut classifier can be improved by adjusting and optimizing the features used to represent symbols. Selecting the best combination of features is not a trivial task, and it has been demonstrated that a high number of features does not necessarily increase the recognition rate [6]. For this reason, we used the same limited set of features as in [6], i.e., *aspect ratio*, *moments*, *nrows*, and *volume64regions*. The genetic algorithm (GA) for feature weight optimization in Gamut was then tested in our experiments. As optimizing a classifier is a fairly computationally expensive task (approximately 24 hours for a minimal optimization of one 40-page classifier), we were not able to test this approach on all generated classifiers.

Similarly, Aruspix HMMs may be improved with the incorporation of *n-gram*-based music models. These models are built by performing statistical analysis on training data, and used during decoding to evaluate the likelihood of symbol classification according to $n-1$ previously recognized symbols. In our experiments, we chose $n = 3$, which conveys that the likelihood of symbols to be recognized is evaluated according to the two preceding symbols. The music models were generated from the same data as from training HMMs (using 1 to 40 pages for each book).

3.3 Post-processing in Gamut: pitch detection

In Gamut, a post-processing step is required to retrieve pitch points from the recognized symbols. This task can be done heuristically, for example, by localizing the center of a note head according to the staff line positions. With well-printed and non-degraded documents, this is usually a straightforward task. Early music sources, however, introduce many printing irregularities and document degradations, which often result in minor failures during staff removal (i.e., small staff elements remaining), making this post-processing step much more challenging. Thus, as the solution to the problem is highly dependent on the type of music documents being processed, we had to design an entire set of specialized heuristic methods to retrieve the pitch points. The heuristic methods are based on basic image analysis techniques such as projection analysis and centroid detection. This was done using the plugins and toolkit facilities in Gamut which enable custom functionalities to be implemented easily and added to the framework.

3.4 Evaluation procedure

The recognition results were evaluated by calculating both recall and precision on the best-aligned sequences of recognized symbols. Other existing evaluation methods such as Bellini et al. [2] were reviewed, but they were not found to be relevant to the present study at it was primarily de-

signed for common music notation. Our first evaluation was to measure the performance of the core recognition engines by determining symbol recognition rates for both the Gamut kNN and Aruspix HMM methods, without pitch detection. This was a trivial alteration within Gamut, as we only needed to consider the output of the classifier, prior to pitch point detection. For Aruspix, however, it was necessary to short-circuit the system, as symbol and pitch recognition is normally performed concurrently. The HMMs were modified in such a way that pitches were ignored, reducing the number of different symbols to be recognized from more than 130 to less than 30, hence enabling a direct comparison with Gamut classifiers. We also evaluated and compared improvement rates of both systems through incorporation of their specific optimization schemes, i.e., the genetic algorithm for Gamut and *n-gram*-based music models for Aruspix. Finally, we assessed the complete recognition process including pitch detection, yielding a full comparison of both OMR approaches for early music prints.

4 RESULTS

4.1 Evaluation of the recognition systems

Training the recognition systems of Gamut and Aruspix neither yields similar results nor generates the same learning curves. After 40 pages of training, Aruspix HMMs outperform the Gamut kNN classifiers for all four books, as shown in table 1. Only within the fourth book (M-0585) does the kNN classifier produce similar results to those of the HMMs, with more than 93% recall and more than 94% precision. Overall, the HMMs appears to be more robust and reliable, as the variability across the results is far smaller than within the kNN classifiers. While learning rates for Aruspix HMMs are fairly consistent across the four books, the Gamut kNN classifiers varies significantly from one book to another.

Figure 3 shows the learning curves for Aruspix HMMs and Gamut kNN classifiers for the book M-0579 and for the book M-0585. When the Gamut kNN classifiers perform well, they learn quicker than the HMMs, as demonstrated within the fourth book M-0585 (figure 3, bottom). In all cases, the kNN classifiers reach a plateau after 10 to 20

Book	Recall		Precision	
	kNN	HMMs	kNN	HMMs
M-0579	86.90	95.99	86.63	95.63
M-0580	86.76	93.51	89.36	97.32
M-0583	77.30	87.58	81.04	93.95
M-0585	93.35	94.72	93.47	96.80

Table 1: Recall and precision for the core recognition systems after 40 pages

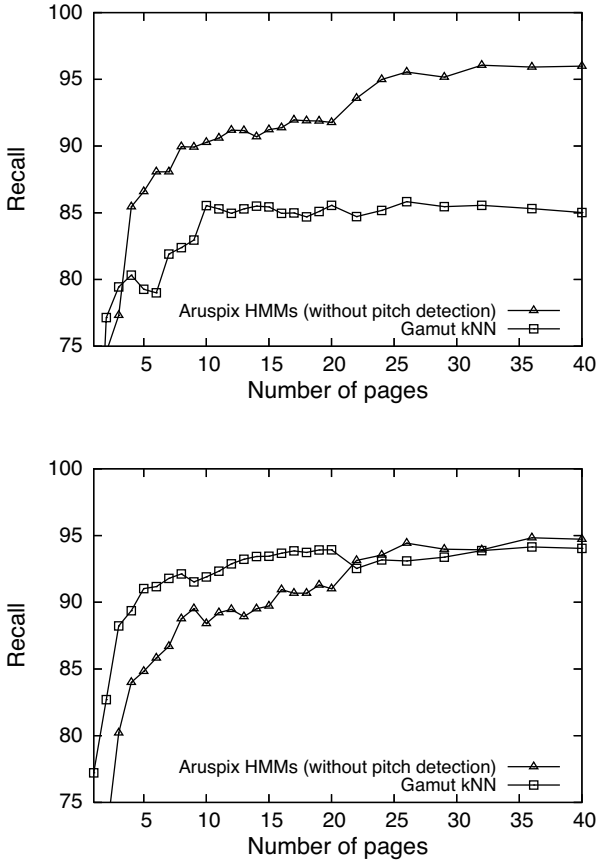


Figure 3: Learning curves (recall) of the core recognition systems on M-0579 and M-0585

pages, whereas the HMMs continue to improve gradually even after 30 pages.

For both Gamut kNN classifiers and Aruspix HMMs, the third book (M-0583) appears to be the most difficult. It is the most degraded book in our set, with strong bleed-through showing printed characters of the verso through the page. This book was printed with a very similar music font to that in book M-0585 (see figure 1), upon which both systems performed quite well: the best results for Gamut were found within this book. We may therefore conclude that document degradation, and not font shape, is the main cause of difficulty for both systems encountered with this book. Such degradations make the binarization step, which is critical in such applications, much more challenging [3]. Figure 4 clearly illustrates the effects of binarization errors on subsequent staff removal and recognition.

4.2 Evaluation of the optimization techniques

GA optimization of Gamut kNN classifiers improved the results for each of the four books in our experiments. Further, it appeared to be extremely influential when the number of

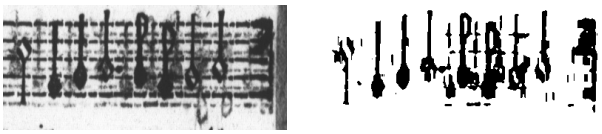


Figure 4: Original greyscale image and the same image after binarization and staff removal in one page of M-0583

pages was very small. Even within kNN classifiers already demonstrating steep (fast) learning curves, the GA optimization led to markedly faster learning, as shown in figure 5. Yet, in most cases, results were not significantly improved when using 20 pages or more.

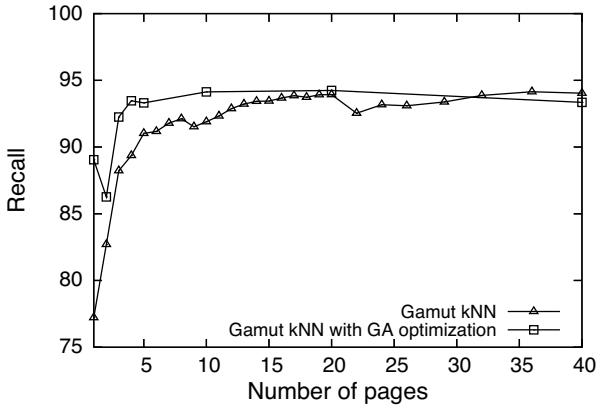


Figure 5: Improvement (recall) with GA optimization in Gamut on M-0585

For Aruspix, the integration of the *n-gram*-based models also improved results for all books, but in a notably different way than GA optimization within Gamut. First, the *n-gram*-based models improve precision rates much more so than recall rates. This verifies the logical intuition that the introduction of the model will reduce the likelihood of non-expected symbols during decoding, and thus reduce the number of insertions. Secondly, the improvement curve is also quite different than that of GA optimization within Gamut, as *n-gram* integration requires more pages to be significant, but displays a longer stable growth period, as shown in figure 6.

4.3 Evaluation of the OMR approaches

Table 2 presents the overall OMR recall and precision rates of both applications after 40 pages. In Gamut, pitches were determined heuristically in a post-processing step, whereas Aruspix uses an all-inclusive approach directly at the HMM level. On average, the differences for the complete OMR process from one book to the next are very similar to the those observed for the core recognition systems (see table 1), with a more significant loss in Gamut than in Aruspix.

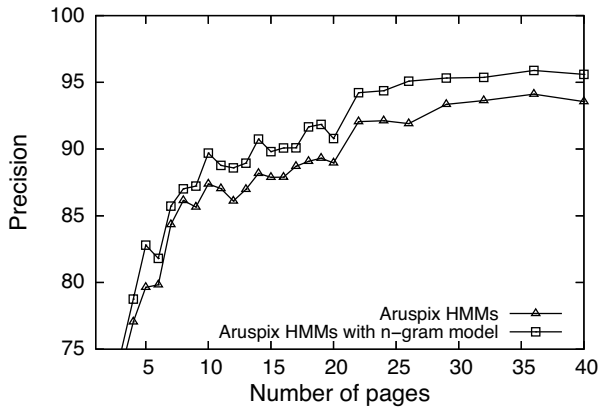


Figure 6: Improvement (precision) with the *n-gram*-based music models in Aruspix on M-0579

Book	Recall		Precision	
	Gamut	Aruspix	Gamut	Aruspix
M-0579	79.34	92.44	84.13	95.60
M-0580	76.18	92.52	84.55	96.44
M-0583	68.13	86.75	76.54	92.76
M-0585	81.61	92.77	88.34	96.23

Table 2: Recall and precision for the complete OMR results after 40 pages

These results distinctly show that retrieval of pitch points after symbol recognition is far from trivial with early music documents. In this task, the flexibility of Gamut could have been leveraged more, as the heuristic approach may be adjusted for a particular dataset.

In our experiment, Gamut failed to perform as well as Aruspix, despite the fact that the all-inclusive approach of Aruspix offer very few entry points for adjustment. Within the fourth book (M-0585), for which the core recognition systems performance is nearly equivalent, the final OMR output decreases significantly with the addition of pitch detection in Gamut (-11.71% for the recall rate) whereas the decline is only minimal in Aruspix (-1.95%). This slight reduction at the incorporation of pitch detection in Aruspix is an intriguing aspect of these results, as the all-inclusive approach of Aruspix has drastically inflated the number of symbols to be recognized (from less than 30 to more than 130 with our datasets) without affecting the recognition rates significantly.

5 CONCLUSIONS AND FUTURE WORK

The results of these experiments confirm the need for adaptability within tools for optical early music recognition. Even if the systems learn and perform differently, they both under-

performed on the same book (M-0585), which was by far the most degraded in our experimental set. This clearly indicates that binarization of degraded music documents is still an area in which improvements must be made. The difference between the results of M-0582 and M-0585, two books with the same content and quasi-identical music font, illustrates well the extent to which degradation may affect the performance of the systems. This is a quite probable explanation as to why our results obtained with Gamut are significantly lower than those obtained by Dalitz et al. [6] from a lute tablature application also built within the Gamut framework, which was evaluated with retouched facsimile images.

Our experiments with Gamut also demonstrate that staff removal within early music sources cannot be considered a solved problem. None of the algorithms tested worked extraordinarily well, and even minor failures, such as remaining small staff elements, certainly affect performance throughout the entire OMR process. Retrieving the pitch from the recognized symbols appears to be a challenging step within such documents, and the heuristic method currently used in Gamut could certainly be improved further.

The flexibility and the extensibility of the Gamut infrastructure proved its utility, as the system can easily be modified to improve its performance for a particular set of sources. Additionally, the learning speed of Gamut is an asset to the approach. The kNN classifiers learn quite rapidly, enabling a specific application to be built upon only a handful of pages. With the addition of the GA optimization, the amount of data required to achieve the best performance is reduced even further. Aruspix distinguishes itself by its performance and robustness. Together, HMMs and integrated pitch detection are an excellent approach with which to handle noise introduced by printing irregularities and degradations in early printed sources. Aruspix also provides a built-in editor specifically designed to assist in the correction of recognition errors. To enhance performance, Aruspix implements dynamic adaptation, which enables the amount of training data to be significantly reduced through optimization of previously trained HMMs.

Although both applications present distinct advantages, Aruspix clearly benefits from having been optimized for early music prints and may be used directly out-of-the-box, while the adaptability of Gamut may be advantageous for projects with a wider scope.

6 REFERENCES

[1] D. Baindridge and T. Bell. The challenge of optical music recognition. In *Computer and the Humanities*, volume 35, pages 95–121, 2001.

[2] P. Bellini, I. Bruno, and P. Nesi. Assessing optical music recognition tools. *Computer Music Journal*, 31(1):68–93, 2007.

[3] J. A. Burgoyne, L. Pugin, G. Eustace, and I. Fujinaga. A comparative survey of image binarisation algorithms for optical recognition on degraded musical sources. In *Proceedings of the 8th International Conference on Music Information Retrieval*, pages 509–12, Vienna, Austria, 2007.

[4] D. Byrd and M. Schindele. Prospects for improving OMR with multiple recognizers. In *Proceedings of the 7th International Conference on Music Information Retrieval*, pages 41–6, Victoria, Canada, 2006.

[5] C. Dalitz, M. Droettboom, B. Czerwinski, and I. Fujinaga. A comparative study of staff removal algorithms. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(5):753–66, May 2008.

[6] C. Dalitz and T. Karsten. Using the Gamera framework for building a lute tablature recognition system. In *Proceedings of the 6th International Conference on Music Information Retrieval*, pages 478–81, London, UK, 2005.

[7] M. Droettboom. Correcting broken characters in the recognition of historical printed documents. In *Proceedings of the 3rd ACM/IEEE-CS Joint Conference on Digital Libraries*, pages 364–6, Houston, Texas, 2003.

[8] M. Droettboom, K. MacMillan, I. Fujinaga, G. S. Choudhury, T. DiLauro, M. Patton, and T. Anderson. Using the Gamera framework for the recognition of cultural heritage materials. In *Proceedings of the 2nd ACM/IEEE-CS Joint Conference on Digital Libraries*, pages 11–7, Portland, Oregon, 2002.

[9] I. Fujinaga. Staff detection and removal. In S. E. George, editor, *Visual Perception of Music Notation: On-Line and Off-Line Recognition*, chapter 1, pages 1–39. IRM Press, Hershey, 2005.

[10] K. MacMillan, M. Droettboom, and I. Fujinaga. Gamera: Optical music recognition in a new shell. In *Proceedings of the International Computer Music Conference*, pages 482–5, 2002.

[11] J. C. Pinto, P. Vieira, M. Ramalho, M. Mengucci, P. Pina, and F. Muge. Ancient music recovery for digital libraries. In *4th European Conference on Digital Libraries, ECDL 2000, Lisbon, Portugal, September 2000, Proceedings*, volume 1923 of *LNCS*, pages 24–34. Springer, Berlin, 2000.

[12] L. Pugin. Optical music recognition of early typographic prints using hidden Markov models. In *Proceedings of the 7th International Conference on Music Information Retrieval*, pages 53–6, Victoria, Canada, 2006.

[13] L. Pugin, J. A. Burgoyne, and I. Fujinaga. Goal-directed evaluation for the improvement of optical music recognition on early music prints. In *Proceedings of the ACM/IEEE Joint Conference on Digital Libraries*, pages 303–4, Vancouver, Canada, 2007.

[14] L. Pugin, J. A. Burgoyne, and I. Fujinaga. MAP adaptation to improve optical music recognition of early music documents using hidden Markov models. In *Proceedings of the 8th International Conference on Music Information Retrieval*, pages 513–6, Vienna, Austria, 2007.

[15] Répertoire international des sources musicales (RISM). *Single Prints Before 1800*. Series A/I. Bärenreiter, Kassel, 1971–81.