Proceedings of the 1st International Workshop on Reading Music Systems

20th September, 2018
Paris, France

Proceedings of the 1st International Workshop on Reading Music Systems, Paris, 2018

Edited by Jorge Calvo-Zaragoza, Jan Hajič jr. and Alexander Pacha

Logo made by Freepik from www.flaticon.com. Adapted by Alexander Pacha

Organization

General Chairs

Jorge Calvo-Zaragoza	Universitat Politècnica de València, Spain
Jan Hajič jr.	Charles University, Czech Republic
Alexander Pacha	TU Wien, Austria

Local Organizer

Philippe Rigaux	Conservatoire national des arts et métiers, France

Program Committee

Ichiro Fujinaga	McGill University, Canada
Alicia Fornés	Computer Vision Center, Spain
Andreas Arzt	Johannes Kepler Universität, Austria
Horst Eidenberger	TU Wien, Austria
Jose M. Iñesta	University of Alicante, Spain
Ana Rebelo	INESC Tec, Portugal
Christopher Raphael	Indiana University, United States

Support

The organization of this workshop has been possible thanks to the support from the Music Notation Information Retrieval (MuNIR) project.

Preface

Dear colleagues,

it is our greatest pleasure to introduce the proceedings of the 1st International Workshop on Music Reading Systems (WoRMS), an event that many of us have been looking forward to for a long time. Considering that Optical Music Recognition (OMR) has been subject to research for over 50 years, it is high time for all active researchers in this field to gather in one place to exchange ideas and build a stable community that will drive the research for the years to come.

At the Graphics Recognition Workshop (GREC) 2017 in Kyoto, many researchers in this field met in person for the first time. It was a unique opportunity that created a wide range of collaborations and scientific exchange. In that spirit, we decided to create a similar event, one that is dedicated entirely to OMR and is attached to the annual International Society for Music Information Retrieval Conference (ISMIR), which feels to us like a natural choice for the venue, considering that OMR lives at the intersection between music and technology and is mainly used by people that are interested in music or music information retrieval.

It is an exciting time to be an active researcher in this field with significant advances happening all around the globe, making it sometimes hard to oversee all of them. We are very happy that the topics of the four sessions span the entire bandwidth: Community, Applications and Interactive Systems, Technical Solutions, and User Perspectives. The brief presentation format with an open-ended discussion towards the end of the workshop will hopefully trigger lively discussions, spawn inspiring ideas and help initiate collaborations that will make this workshop a memorable event.

Jorge Calvo-Zaragoza, Jan Hajič jr. and Alexander Pacha

Contents

A Starting Point for Handwritten Music Recognition

Arnau Baró, Pau Riba and Alicia Fornés
Computer Vision Center - Computer Science Department
Universitat Autònoma de Barcelona, Bellaterra, Catalonia, Spain
Email: {abaro,priba,afornes}@cvc.uab.cat

***Abstract*—In the last years, the interest in Optical Music Recognition (OMR) has reawakened, especially since the appearance of deep learning. However, there are very few works addressing handwritten scores. In this work we describe a full OMR pipeline for handwritten music scores by using Convolutional and Recurrent Neural Networks that could serve as a baseline for the research community.**

***Index Terms*—Optical Music Recognition, Long Short-Term Memory, Convolutional Neural Networks, MUSCIMA++, CVC-MUSCIMA.**

I. Introduction

For many decades, music scores have been manually written in a sheet format. Nowadays, there are archives with thousands of handwritten music scores waiting to be transcribed. Since a manual transcription becomes unfeasible, it is necessary to develop an automatic method to transcribe music scores.

Optical Music Recognition (OMR) is the process to convert a music score image into a machine-readable format. There are some OMR software such as PhotoScore [1] or SharpEye[2] that work very well on printed scores. However, when they have to deal with handwritten music scores, the accuracy decreases significantly. As far as we know, the few existing OMR handwritten methods only focus on a specific stage of the full OMR pipeline, such as layout analysis [1], detection and classification of graphic primitives [2], [3]. Thus, we believe that it is time to design a full OMR pipeline for handwritten scores.

In this paper we propose a full staff-wise OMR system for handwritten music scores which can serve as a baseline for the research community. Our method is composed by a Convolutional Neural Network followed by a Bidirectional Long Short-Term Recurrent Neural Network. This architecture is based on our previous publication [4], where we proposed to recognize printed music scores as a sequential recognition task by using Bidirectional Long Short-Term Memory networks. In the current work we improve and adapt this architecture for dealing with handwritten music scores. First, we have added a Convolutional Neural Network in order to extract meaningful features. Since the amount of annotated handwritten data is very limited, we propose a specific data augmentation technique to increase the amount of training music scores. Finally, we have studied transfer learning techniques to benefit from printed and synthetic music scores.

[1]http://www.neuratron.com/photoscore.htm
[2]http://www.visiv.co.uk/

II. Proposed architecture

Single staff sheet music can be seen as a sequence. In this way, first of all we have cropped the different staves of each page to read them from left to right. Then, our architecture is composed of the following steps:

Input: Each music score is first preprocessed cropping it into staves and then each staff is resized to a height of 100 pixels and binarized to feed each column in the proposed architecture. Since the aspect ratio is kept, each image will have different width. Afterwards, all images belonging to the same batch are padded to the maximum width.

Convolutional Block: This block uses three convolutional layers with a kernel of 3x3 which increases the depth. Batch Normalization and Rectified Linear Unit activations are located after each convolutional layer. Then a max-pool 2x1 operator is used to maintain the image width in order to feed our model column-by-column.

Recurrent Block: This block is constructed by 4 Bidirectional Long Short-Term Memory (BLSTM) layers of 512 neurons each. The bidirectionallity provides more context than a single LSTM, because ambiguities can be reduced when taking into account the forward and backward directions in the image. For example, if one direction is reading a vertical line and the other direction is seeing a notehead, the network can correctly predict a quarter note.

Dense Layers: After the recurrent block, two fully connected (FC) layers are located. We use two separated fully connected layers in order to reduce the large combination between rhythm and melody. Beside this, by separating rhythm and melody, the system is able to learn the shape of a symbol independently of its position on the staff and vice versa. In other words, the system can learn the shape of a quarter note no matter if it is located in the first or fourth line in the staff.

Output: Lastly, each fully connected layer returns a binary matrix. Each matrix has the same width as the original image. The rhythm is defined using a matrix with a height of 80 classes and the pitch with 28 classes. Finally, a threshold is applied to decide which symbols appear in the music score.

Both matrices are finally converted into one array. So, we obtain three arrays, one for the rhythm, other for the pitch and another one with the combination of both. These arrays will be used to evaluate the method.

III. Training strategies

Since there are very few labelled handwritten music scores to train the systems, this leads to overfitting problems. For this

TABLE I
RESULTS. SYMBOL ERROR RATES ARE BETWEEN [0-1], AND GIVEN BY THE MEAN OF FIVE EXECUTIONS.

Pre-train Printed	D. Augm. Printed	BLSTM	CNN	D. Augm. Handwritten		Rhythm SER	Pitch SER	Rhythm+Pitch SER
				Shuffle	Morph.			
-	-	-	-	-	-	0.826	0.709	0.899
✓	-	-	-	-	-	0.771	0.668	0.872
✓	✓	-	-	-	-	0.762	0.690	0.854
✓	✓	✓	-	-	-	0.523	0.464	0.610
✓	✓	✓	✓	-	-	0.493	0.396	0.559
✓	✓	✓	✓	✓	-	**0.476**	**0.387**	**0.545**
✓	✓	✓	✓	✓	✓	0.490	0.393	0.554

reason, we opt to use printed data as pre-training.

Concretely, we propose to train our model with printed data and afterwards retrain it with the few available handwritten data. In other words, we propose transfer learning by fine-tuning the pre-trained system with the handwritten data.

In addition, some distortions have been applied to the training data so that the system can learn a more robust and general shape of every symbol. First, we have applied three distortion methods -dilating, eroding or blurring- into both the printed and handwritten training sets. Secondly, and in order to increase the amount of possible melodies, the number of handwritten training music scores has been increased by cropping each measure (bar unit) and shuffling them in the music score.

IV. EXPERIMENTATION

To evaluate our OMR system on handwritten music scores, we have labelled at symbol level a subset of 20 pages of the MUSCIMA++ dataset [5], which is a subset of the CVC-MUSCIMA dataset [6]. For measuring the performance, we have used the Symbol Error Rate (SER) as evaluation metric, defined as the sum of edit operations that are needed to convert the output of our architecture into the label in terms of symbol insertions, substitutions and deletions.

Table I shows the results of our method. In the table, the lower SER, the better. Note that each line introduces an improvement to the previous system. In the first row we are testing our method without any of the proposed improvements nor pre-training on printed data.

It can be observed that the incorporation of the data augmentation on the printed dataset, and shuffling the measures indeed improves the overall system performance. Also, instead of using the raw image as input, the convolutional block shows that it indeed helps to extract the discriminative features to recognize the different music symbols. Nevertheless, the BLSTM is the key modification to reduce the error rates by a large margin. This is because the bidirectionality is able to reduce most of the ambiguities. Finally, note that using morphological operations for data augmentation only introduces noise and increases the error rates.

V. CONCLUSIONS AND FUTURE WORK

In this work, we have proposed a full Optical Music Recognition system to recognize handwritten music scores. This method uses a convolution block to extract features followed by a recurrent block based on the Bidirectional Long Short-Term Memory Recurrent Neural Networks. It also includes a specific data augmentation and uses transfer learning from printed data.

We believe that the provided results can be used as a baseline for future improvements in the OMR research field. Bearing in mind that we have only used 20 pages of the MUSCIMA++, these results are encouraging. We think that increasing the amount of annotated handwritten music scores at symbol level, the system could obtain much better results.

Future work will be focused on investigating more suitable data augmentation methods for music scores. Moreover, we would like to study how a postprocessing step based on grammars or semantics could solve ambiguities at higher level, and thus improve the performance.

ACKNOWLEDGMENTS

This work has been partially supported by the project TIN2015-70924-C2-2-R, the CERCA Program/Generalitat de Catalunya, and the Fellowships AGAUR 2018 FI_B 00546, FPU15/06264 and RYC-2014-16831.

REFERENCES

[1] J. Calvo-Zaragoza, F. J. Castellanos, G. Vigliensoni, and I. Fujinaga, "Deep neural networks for document processing of music score images," *Applied Sciences*, vol. 8, no. 5, pp. 654–674, 2018.

[2] J. Hajič jr. and P. Pecina, "Detecting noteheads in handwritten scores with convnets and bounding box regression," *CoRR*, vol. abs/1708.01806, 2017.

[3] L. Tuggener, I. Elezi, J. Schmidhuber, and T. Stadelmann, "Deep watershed detector for music object recognition (accepted)," in *ISMIR*, 2018.

[4] A. Baró, P. Riba, J. Calvo-Zaragoza, and A. Fornés, "Optical music recognition by long short-term memory networks," in *Graphic Recognition. Current Trends and Challenges*, 2018.

[5] J. Hajič jr. and P. Pecina, "The MUSCIMA++ Dataset for Handwritten Optical Music Recognition," in *ICDAR*, 2017, pp. 39–46.

[6] A. Fornés, A. Dutta, A. Gordo, and J. Lladós, "CVC-MUSCIMA: a ground truth of handwritten music score images for writer identification and staff removal," *International Journal on Document Analysis and Recognition (IJDAR)*, vol. 15, no. 3, pp. 243–251, 2012.

Why WoRMS?

Jorge Calvo-Zaragoza
PRHLT Research Center
Universitat Politècnica de València
Valencia, Spain
jcalvo@prhlt.upv.es

***Abstract*—In this paper the organization of the first Workshop on Reading Music Systems (WoRMS) is motivated. The event is intended to become a forum for discussion and collaborative projects for all people interested in the optical music recognition and related applications. This first edition is held as a satellite event of the 19th International Society for Music Information Retrieval Conference. It is our hope that WoRMS helps to set the basis towards building a stable community around the field of music reading systems.**

***Index Terms*—Optical Music Recognition, Automatic Music Reading, Music Information Retrieval, Music Digital Libraries**

I. Introduction

Optical music recognition (OMR) is the research field that investigates how to make computers be capable of reading music. It is a very attractive field from different points of view. The obvious one is the hope of providing an efficient way of converting written music to a symbolically encoded format. On the other hand, the field is exciting for research from the perspective of computer science, machine learning and pattern recognition: while it might be true that the task of reading music computationally shares similarities with others (for instance, the optical recognition of text), music notation has enough nuances that make it appropriate to be considered as a specific field [1], and whose specific research cannot be easily generalized from other domains.

Despite all the above, OMR has experienced a slow progress. One of the main reasons for such a troublesome development is that there are few people devoted to working it out. Although the different perspectives of OMR make it really interesting for general research, it is difficult to find single specialists that cover all related areas (image processing, machine learning, computer music, and music notation in general). Perhaps, people who do not meet all these requirements feel some sense of vertigo before facing the problem, thereby choosing another field of research.

More importantly, we believe that there is a more relevant obstacle preventing a proper development of the field: an evident lack of communication in the community. Over the years, many techniques have been proposed to deal with OMR-related problems [2], but there are hardly re-uses of the work done previously by other researchers. This might be caused by the heterogeneity inherent to the field: publications in OMR are rather scattered. The purely methodological aspect is more related to machine learning, pattern recognition and document analysis—and there are many journals and conferences on these topics; however, the music information retrieval and digital libraries communities are the ones especially interested in OMR results, and so many publications have also fallen in this type of venues. Consequently, the different OMR authors rarely meet in person.

And it is for all the above, that the OMR community can hardly set the basis of the field so that researchers do not need to reinvent the wheel over and over again. There have been so many different approaches, data formats, evaluation criteria, and datasets that a more formal development of the field is somehow difficult to attain. That is why the organization of an event about optical recognition technologies for music notation becomes necessary.

II. A Workshop on Music Reading Systems

The 12th IAPR International Workshop on Graphics Recognition (GREC'17), co-located with the 14th IAPR International Conference on Document Analysis and Recognition (ICDAR'17), gathered a number of authors working in OMR (including the keynote!).[1] In spite of some notable absences, it was a unique opportunity to hold a discussion group entirely dedicated to OMR [3]. However, it is important to note that this concentration of OMR researchers was thanks to the efforts of one of the organizers of GREC'17 (Alicia Fornés), who personally contacted many of these researchers to make contributions to the workshop.

During the discussion session, the latest contributions in the field were discussed: deep learning, as in other communities, provides a way of training powerful and accurate models, and conventional tasks (such as symbol classification or staff-line removal) no longer represent actual challenges [4], [5]. In fact, there is a current trend towards systems that decompose the recognition workflow in fewer stages, such as the direct detection of musical objects in the images [6] or end-to-end systems [7], [8]. In addition, public datasets recently released, such as MUSCIMA++ [9], PrIMuS [10] or DeepScores [11], allow researchers to perform comprehensive and reproducible experiments. Furthermore, given the complexity of the recognition task—which makes it difficult to think of achieving perfect results—another prominent line of current research is about

This work is supported by the Spanish Ministerio de Economía, Industria y Competitividad through Juan de la Cierva - Formación grant (Ref. FJCI-2016-27873).

[1] http://grec2017.loria.fr

interactive approaches [12], [13], where the human-machine interaction plays a key role in the workflow. Nevertheless, one issue upon which all the participants in the discussion agreed was the lack of adequate evaluation measures [14], [15], which indeed points to incomplete formulations of the OMR field in terms of input-output tasks.

The main conclusion of the discussion group was that OMR needs to properly define itself. This means finding proper formulations of its associated tasks, along with their corresponding evaluation criteria, that can be shared by the entire community. Hence, the idea of organizing an event exclusively dedicated to the technologies of reading systems for music notation naturally arose.

As mentioned on the website,[2] the international Workshop on Reading Music Systems (WoRMS) is "*a novel workshop that tries to connect researchers who develop systems for reading music, such as in the field of Optical Music Recognition, with other researchers and practitioners that could benefit from such systems, like librarians or musicologists*".

It is worth emphasizing that the workshop is about systems that read documents of written music, which is not strictly exclusive of OMR: tasks such as score following or palaeographic analysis do have room in WoRMS, as long as the objects of study are related to this kind of documents. Furthermore, the workshop explicitly welcomes people without a technical background, yet interested in either using the technology or carrying out focused projects.

This first edition of the event is held as a satellite event of the 19th International Society for Music Information Retrieval Conference (ISMIR'18). Given that there exist several events about machine learning applied to document analysis (such as the International Conference on Document Analysis and Recognition and the International Conference on Frontiers in Handwriting Recognition), we believe that it is more interesting to get WoRMS closer to the community where applications may find real use, namely the Music Information Retrieval one. If the idea is to define the shape of the field, we definitely need to know how to formulate the tasks according to end users' needs, and therefore celebrating the event within ISMIR'18 may bring this type of audience as well.

III. Conclusions

Technology for the development of music reading systems is an exciting research avenue with a number of potential applications. However, the field might not have sufficient maturity due to the lack of a communicative community. This prevents the proper establishment of task formulations, evaluation protocols and generalizable techniques that can be exploited by other researchers.

The organization of WoRMS is a historic opportunity to set the basis towards an agreement on these issues, combining both the technical developments and the specific aspects of the different applications. We hope WoRMS will provide an opportunity to share ideas and discuss about current issues, with the aim of building a stable community around the field of music reading systems.

[2]https://sites.google.com/view/worms2018

Acknowledgements

I would like to thank my colleagues Jan Hajič jr. and Alexander Pacha for sharing the excitement of organizing this event. We do acknowledge the support from Ichiro Fujinaga and Philippe Rigaux, who helped us to make WoRMS become real.

References

[1] D. Bainbridge and T. Bell, "The challenge of optical music recognition," *Computers and the Humanities*, vol. 35, no. 2, pp. 95–121, 2001.

[2] A. Rebelo, I. Fujinaga, F. Paszkiewicz, A. R. S. Marçal, C. Guedes, and J. S. Cardoso, "Optical music recognition: state-of-the-art and open issues," *International Journal of Multimedia Information Retrieval*, vol. 1, no. 3, pp. 173–190, 2012.

[3] J. Calvo-Zaragoza, J. Hajič jr., and A. Pacha, "Optical music recognition discussion group at the graphics recognition workshop 2017," in *12th International Workshop on Graphics Recognition. Revised Selected Papers*, 2018, (in press).

[4] A.-J. Gallego and J. Calvo-Zaragoza, "Staff-line removal with selectional auto-encoders," *Expert Systems with Applications*, vol. 89, pp. 138–148, 2017.

[5] A. Pacha and H. Eidenberger, "Towards a universal music symbol classifier," in *Proceedings of the 12th IAPR International Workshop on Graphics Recognition*. IEEE Computer Society, 2017, pp. 35–36.

[6] A. Pacha, K.-Y. Choi, B. Coüasnon, Y. Ricquebourg, R. Zanibbi, and H. Eidenberger, "Handwritten music object detection: Open issues and baseline results," in *2018 13th IAPR Workshop on Document Analysis Systems (DAS)*, 2018, pp. 163–168.

[7] E. van der Wel and K. Ullrich, "Optical music recognition with convolutional sequence-to-sequence models," in *Proceedings of the 18th International Society for Music Information Retrieval Conference*, 2017, pp. 731–737.

[8] J. Calvo-Zaragoza and D. Rizo, "End-to-end neural optical music recognition of monophonic scores," *Applied Sciences*, no. 4, p. 606, 2018.

[9] J. Hajic and P. Pecina, "The MUSCIMA++ dataset for handwritten optical music recognition," in *14th IAPR International Conference on Document Analysis and Recognition*, 2017, pp. 39–46.

[10] J. Calvo-Zaragoza and D. Rizo, "Camera-PrIMuS: Neural end-to-end optical music recognition on realistic monophonic scores," in *19th International Society for Music Information Retrieval Conference*, 2018, (in press).

[11] L. Tuggener, I. Elezi, J. Schmidhuber, M. Pelillo, and T. Stadelmann, "DeepScores - A Dataset for Segmentation, Detection and Classification of Tiny Objects," *Computer Research Repository*, vol. abs/1804.00525, 2018.

[12] L. Chen, E. Stolterman, and C. Raphael, "Human-interactive optical music recognition," in *Proceedings of the 17th International Society for Music Information Retrieval Conference*, 2016, pp. 647–653.

[13] G. Vigliensoni, J. Calvo-Zaragoza, and I. Fujinaga, "An environment for machine pedagogy: Learning how to teach computers to read music," in *Joint Proceedings of the ACM IUI 2018 Workshops co-located with the 23rd ACM Conference on Intelligent User Interfaces*.

[14] D. Byrd and J. G. Simonsen, "Towards a standard testbed for optical music recognition: Definitions, metrics, and page images," *Journal of New Music Research*, vol. 44, no. 3, pp. 169–195, 2015.

[15] J. Hajič jr., J. Novotný, P. Pecina, and J. Pokorný, "Further steps towards a standard testbed for optical music recognition," in *Proceedings of the 17th International Society for Music Information Retrieval Conference*, 2016, pp. 157–163.

Music Symbol Detection with Faster R-CNN Using Synthetic Annotations

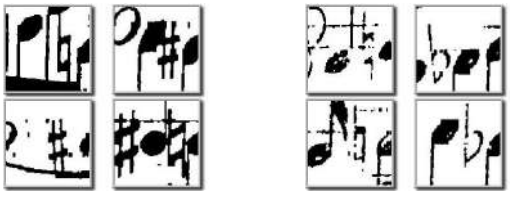
Kwon-Young Choi, Bertrand Coüasnon, Yann Ricquebourg
Univ Rennes, CNRS, IRISA, F-35000
Rennes, France
{kwon-young.choi, bertrand.couasnon, yann.ricquebourg}@irisa.fr

Richard Zanibbi
Rochester Institute of Technology
Rochester, USA
rlaz@cs.rit.edu

***Abstract*—Accurately detecting music symbols in images of historical, complex, dense orchestral or piano printed scores can be challenging due to old printing techniques or time degradations. Because segmentation problems can vary widely, a data driven approach like the use of deep learning detectors is needed. However, the production of detection annotations (symbol bounding boxes + classes) for such systems is costly and time consuming. We propose to train such model with synthetic data and annotations produced by a music typesetting program. We analyze which classes are relevant to the detection task and present a first selection of music score typesetting files that will be used for training. To evaluate our model, we plan to compute quantitative results on a synthetic test set and provide qualitative results on a few manually annotated historical music scores.**

***Index Terms*—Optical Music Recognition, Deep Learning, Symbol Detection**

Introduction: Music symbol detection in historical printed orchestral and piano scores is a challenging problem because of the complexity, density and degradations often present in those scores. Segmentation problems caused by lack of space and time degradations can vary widely as shown in Figure 1. Instead of resolving the segmentation task in a manual fashion with expert knowledge, a better solution is to use trainable models and recent advance in the field of deep learning detectors makes them suitable for this task.



(a) Touching symbols (b) Broken symbols

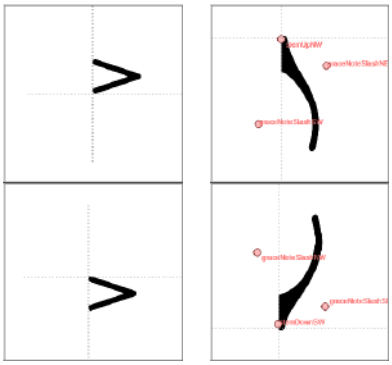
Fig. 1: Common segmentation challenges occurring in real historical piano and orchestral music scores because of engraving or degradation problems.

Previous work [1] has shown that deep learning detectors such as the Faster R-CNN can accurately detect handwritten music symbols. However, the production of annotations for such systems is costly and time consuming. We propose to train such model with synthetic data and annotations produced by a music typesetting software such as MuseScore. Using synthetic data for training machine learning models is a well known subject, as shown by the work of [2] which is able to produce historical synthetic document. However, the use of synthetic data and annotation for the detection of music symbols has still not been applied to the field of Optical Music Recognition. Our end goal is to apply a detector trained only on synthetic data on real historical music scores.

MuseScore Synthetic Data Generation: MuseScore [3] is an open-source music typesetting program that has recently developed a branch for generating data annotations suitable for training classification and detection models used in Optical Music Recognition (OMR) pipeline. Each page of a music score is transformed into a pair of files: an image and an XML files containing a list of symbols present in the image. All symbols are annotated with their class and bounding box in the form of the top-left coordinate and width/height of the symbol. Symbols can be nested, meaning that they are a composite of other symbols or primitives. For example, a nested symbol like the grace note contains elements like a flag, stem, notehead and possible slash, all of which are annotated in the produced XML file with their respective bounding boxes. This opens the possibility of training classifiers and detectors on synthetic data while having lots of flexibility on the class set and composition used. The only limitation of the class set used is imposed by the Standard Music Font Layout (SMuFL) [4] which defines glyphs used in music typesetting software.

Class Set Selection: We first limit the number of classes to the minimal amount possible and group visually similar symbols in the same class. We use these principles to ease the task difficulty for the detector. SMuFL defines all glyphs used to typeset music scores. This standard is diverse and contains around 2600 glyphs. The organization of glyphs are mainly based on their semantic and contextual use. This means that this standard can contain multiple glyphs with the same visual appearance but with only a minor transformation: translation, symmetry or scale as shown in Figure 2.

In music notation, some symbols are built from a number of simple primitives, e.g. flags, rests or dynamics. Using the SMuFL standard, these symbols cannot be decomposed into primitives as they are defined as a single glyphs. Therefore, we choose to use a different class for each different flag, rest and dynamic symbol up to the *flag64th*, *rest64th* and third level of dynamic like *dynamicFFF*. While dynamic symbols cannot be decomposed into their letter primitives, constructs like time



(a) Translation (b) Symmetry

Fig. 2: Visually similar glyphs with different classes in SMuFL. 2a shows symbols *articAccentAbove/Below* which are the same $>$ symbol translated. 2b shows *flag8th* up and down produced with a symmetry.

signature can be decomposed into single digits. We choose to define nine classes corresponding to the different digits from 0 to 9. These digits can be used for the time signature, but also for finger annotation in piano scores.

Very small and complex constructs like grace notes are left for future work as their small size and complexity could be too difficult to handle for a detector with a big input image size.

We also do not consider variable sized symbols like beam, tie, slurs or barline because of their extreme varying size ratio and very simple shapes. Models like line detection models should be more suitable in order to recognize these symbols.

Finally, we propose to leave as future work the definition of meta classes like *timeSig12over8* which annotates a time signature composed of stacked digits 1, 2 and 8 with a global bounding box around the three symbols. Again, this choice is done in order to simplify the task of the detector. Furthermore, meta classes can also be recognized during downstream OMR steps using a contextual approach and a syntactical method.

Using this set of guidelines, we select a set of 55 classes covering most commonly used symbols in orchestral and piano scores, see Figure 3 for an overview of the most common music symbols.

Fig. 3: Common music symbols chosen in the class set composed of accidentals, clefs, ornaments, dynamic signs, tuplets and note heads.

Data Augmentation: In association with the annotation, MuseScore produce an image for each page of a music score. However, these images will be clean, without any kind of noise or deformation. Because our end goal is to use our trained model on real historical scores, we plan to apply common noise and deformation with an open-source software called DocCreator [2]. Noise and deformations can be applied to documents while keeping the spatial annotations like bounding boxes synchronized with the modified document.

Detection Model: For our detection model, we use a regular Faster R-CNN as presented and implemented by [5]. The Faster R-CNN is a two stage detector using a region proposal network (RPN) and a region classifier. The RPN is trained in order to predict possible regions containing an object. The region classifier predicts the class of the object contained in a proposed region while also refining its bounding box. The transition between the RPN and region classifier is done by using an ROI pooling operation which is able to crop a sub-region of the RPN output feature map using the bounding boxes produced by the RPN.

In order to feed the music score to the network, we will first reuse the same strategy as [1] and crop the music score along the stafflines. However, we would like to expand this input size to its maximum and eventually evaluate the performance of the detector applied to a whole page of a music score.

Dataset Description: The dataset is constituted by searching the MuseScore database for scores matching the creative-common zero (CC0) license (equivalent to public domain license). We refine our search by filtering composer known for classical or romantic music like Mozart, Vivaldi, Beethoven or Haydn. Our training dataset is constituted of 48 scores, producing a total of 636 pages and 278797 symbols with 46 different classes. Some previously considered classes like the ‘fermata‘ or keyboard specific symbols are missing and we are looking into expanding our current dataset in order to cover more symbols. We split our dataset at the page level into a train and test set, keeping 70% for training and 30% for testing.

Quantitative and Qualitative Results: We plan to report quantitative results in term of mean Average Precision on a synthetic test set, but also produce some qualitative results on a few *real* printed historical score images. We hope that these results will show that a deep learning detector is able to transfer its learned knowledge from synthetic music scores to real historical music scores.

References

[1] A. Pacha, K.-Y. Choi, B. Couasnon, Y. Ricquebourg, R. Zanibbi, and H. Eidenberger, “Handwritten Music Object Detection: Open Issues and Baseline Results.” IEEE, pp. 163–168.

[2] DocCreator an Application to generate sythetic document images for performance evaluation and retrainning. [Online]. Available: http://doc-creator.labri.fr/

[3] [Online]. Available: https://github.com/musescore/MuseScore

[4] SMuFL. [Online]. Available: https://www.smufl.org/

[5] J. Huang, V. Rathod, C. Sun, M. Zhu, A. Korattikara, A. Fathi, I. Fischer, Z. Wojna, Y. Song, S. Guadarrama, and K. Murphy, “Speed/accuracy trade-offs for modern convolutional object detectors.” [Online]. Available: http://arxiv.org/abs/1611.10012

Optical Music Recognition and Human-in-the-loop Computation

1nd Liang Chen
School of Informatics, Computing and Engineering
Indiana University Bloomington
Bloomington, USA
chen348@indiana.edu

2st Christopher Raphael
School of Informatics, Computing and Engineering
Indiana University Bloomington
Bloomington, USA
craphael@indiana.edu

***Abstract*—We present our work in optical music recognition in which we seek to transform scanned music notation images into symbolic representations. While music notation contains a small core of symbols and primitives composed in a rule-bound way, there are a great many common exceptions to these rules, as well as a heavy tail of rarer symbols. Since our goal is to create symbolic representations with accuracy near that of published music scores, we doubt the feasibility of fully-automatic recognition, opting instead for a human-interactive approach. We define a simple communication channel between the user and recognition engine, in which the user imposes pixel-level or model-level constraints, to improve our automatic OMR system.**

***Index Terms*—Optical Music Recognition, Human-in-the-loop Computation**

I. Introduction

Optical Music Recognition (OMR) seeks to convert music score images into symbolic representations. Success with OMR would pave the way for large symbolic libraries containing all the world's public domain music, that could be instantly accessed, searched, transformed, and reformatted. Such libraries would provide a greatly improved experience for musicians through digital music stands; it would serve as the backbone for developing academic fields, such as computational musicology; and enable emerging applications fusing music and computation such as data-driven composition systems, musical accompaniment systems, and automatic music transcription.

OMR research dates back to the 1960s with mostly-disconnected approaches to many aspects of the problem [1]–[3] including several overviews [4], [5], and well-established commercial systems [6], [7]. In spite of these efforts there is still much to accomplish before the sought-after large-scale symbolic libraries will be in reach. The reason is simply that OMR is *hard*, constituting, in our view, one of the grand challenges of document recognition.

A. Challenges of Optical Music Recognition

Part of OMR's difficulty lies in the high degree of necessary recognition accuracy. The future's digital music stands will require accuracy at least as good as the familiar published hard-copy scores they will displace, otherwise this new technology will not be embraced.

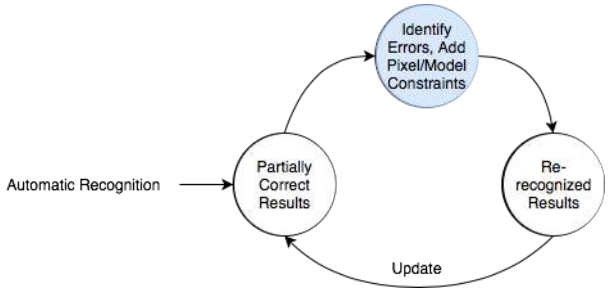


Fig. 1. System design of our human-interactive OMR system. The blue circle corresponds to the human work in the system.

In addition to the high bar regarding quality, OMR poses several significant technical challenges:

- The *two-dimensional* music layout is complex and hard to be analyzed, posing a harder problem than the text recognition.
- The music symbols are constructed from some basic grammatical rules, but these rules are sometimes violated.
- There is a *heavy tail* of standard music symbols. The symbols on the heavy tail are rare enough that their inclusion often results in more false positives than correct detections, yet they must be recognized.

B. Human-in-the-loop Computation

Given the demand for high accuracy and the technical challenges mentioned, we are skeptical that any fully automatic OMR approach will ever deal effectively with the wide range of situations encountered in real-life recognition scenarios. We formulate the challenge as one of *human-in-the-loop computation* instead of a fully automatic one, which fuses both human and machine abilities.

Our essential idea is to allow the user two axes of control over the recognition engine [8], [9]. In one axis the user chooses the *model* that can be used for a given recognition task, specifying both the exceptions to the symbols' construction rules, as well as the relevant variety of symbols to be used. In the other, the user labels misrecognized pixels with the correct primitive type, allowing the system to re-recognize subject to these user-imposed constraints. This approach results in a simple interface in which the user can provide a

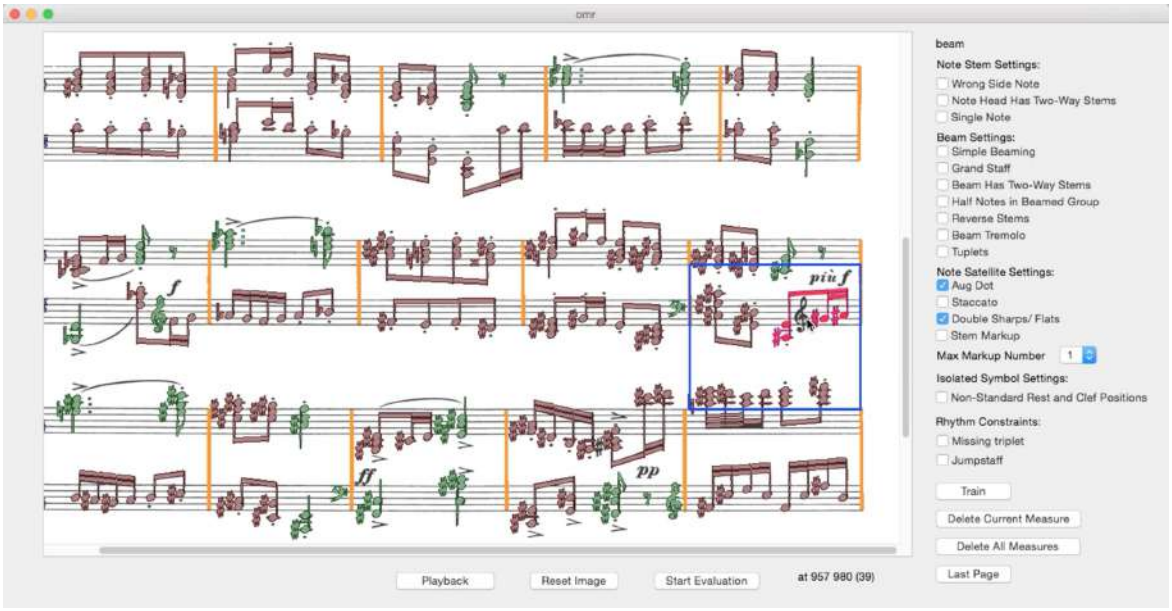


Fig. 2. Interface of our human-interactive OMR system.

wealth of useful knowledge without needing to understand the inner-workings and representations of the system. Thus we effectively address the *communication* issue between human and recognition system.

II. Human-Interactive System, Performance and Conclusion

An overview of our system is illustrated in Fig. 1, where the human action is highlighted in the blue circle. The system is primed with the automatic symbol recognition, after which the system accepts the feedback from the human proofreaders and use it to automatically improve the recognition results. A screenshot of the system interface is shown in Fig. 2. In this example, the user is adding a pixel label for the system to re-recognize the missing small clef. The checking boxes and pull-down menu allows the user to change the model constraints.

We compare the performance of our system against one of the state-of-the-art commercial OMR systems *SmartScore*. While we cannot directly measure intermediate results, SmartScore's raw recognition accuracy appears to be significantly better than ours (at present). However, SmartScore takes less care with the correction of symbols, which holds the system back in an important way. That is, this system appears to be conceived primarily in terms of automatic recognition, with an afterthought that allows the user to correct individual errors. In contrast, our system was conceived as a human-in-the-loop system.

For both systems we concentrate on what happens after automatic recognition, focusing exclusively on the human effort necessary to correct the results. From the experimental results [10], we conclude that our system demonstrates performance that is competitive with SmartScore, in terms of both accuracy and efficiency. The data suggest that SmartScore was slightly more accurate while our system was slightly faster – users make different tradeoffs between these two objectives. In addition, it is worth noting that the data produced by our system also preserves more information about the precise construction and location of image symbols. While beyond the scope of our evaluation, such information can be integral to renotation [11], [12] approaches that leverage specific layout information from the original when creating newly formatted music notation.

References

[1] I. Fujinaga, "Optical music recognition system which learns," in *Proceedings of the SPIE - The International Society for Optical Engineering*, vol. 1785, 1993, pp. 210–17.
[2] G. E. Kopec, P. A. Chou, and D. A. Maltz, "Markov source model for printed music decoding," *Journal of Electronic Imaging*, vol. 5, no. 1, pp. 7–14, 1996.
[3] H. Fahmy and D. Blostein, "A graph-rewriting paradigm for discrete relaxation: Application to sheet-music recognition," *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 12(6), pp. 763–799, 1998.
[4] D. Blostein and H. S. Baird, "A critical survey of music image analysis," in *Structured Document Image Analysis*, 1992, pp. 405–434.
[5] A. Rebelo, G. Capela, and J. S. Cardoso, "Optical recognition of music symbols," *International Journal on Document Analysis and Recognition*, vol. 13, pp. 19–31, 2009.
[6] SmartScore, www.musitek.com, 1991.
[7] SharpEye, www.music-scanning.com/sharpeye.html, 2008.
[8] L. Chen and C. Raphael, "Human-directed optical music recognition," *Electronic Imaging*, vol. 2016, no. 17, pp. 1–9, 2016.
[9] L. Chen, E. Stolterman, and C. Raphael, "Human-interactive optical music recognition." in *ISMIR*, 2016, pp. 647–653.
[10] L. Chen, R. Jin, and C. Raphael, "Human-guided recognition of music score images," in *Proceedings of the 4th International Workshop on Digital Libraries for Musicology*. ACM, 2017, pp. 9–12.
[11] L. Chen, R. Jin, and C. Raphael, "Renotation from optical music recognition," in *International Conference on Mathematics and Computation in Music*, 2015, pp. 16–26.
[12] L. Chen and C. Raphael, "Renotation of optical music recognition data," in *SMC*, 2017.

DeepScores and Deep Watershed Detection: current state and open issues

Ismail Elezi *
ZHAW Datalab
Winterthur, Switzerland
elez@zhaw.ch

Lukas Tuggener *
ZHAW Datalab
Winterthur, Switzerland
tugg@zhaw.ch

Marcello Pelillo
Ca' Foscari University
Venice, Italy
pelillo@unive.it

Thilo Stadelmann
ZHAW Datalab
Winterthur, Switzerland
stdm@zhaw.ch

***Abstract*— This paper gives an overview of our current Optical Music Recognition (OMR) research. We recently released the OMR dataset *DeepScores* as well as the object detection method *Deep Watershed Detector*. We are currently taking some additional steps to improve both of them. Here we summarize current and future efforts, aimed at improving usefulness on real-world task and tackling extreme class imbalance.**

I. INTRODUCTION

The accurate localization and classification of musical symbols is a key component in every functioning Optical Music Recognition (OMR) system [1]. In pursuit of our goal of advancing the state of the art in optical music symbol detection we have created the large *DeepScores* [2] dataset of synthetic music scores together with ground truth to enable the training of very deep neural networks. Additionally, we created a custom object detection method called Deep Watershed Detection [3], that is designed to work particularly well on optical music notation data. Both these contributions currently carry some drawbacks and flaws that hamper performance and usability. In this paper, we give an overview of our current as well as planned efforts to alleviate these issues.

II. UPDATES TO THE *DeepScores* DATASET

A. Shortcomings of the initial release

At its initial release, *DeepScores* had two main weaknesses: first, it was fully geared towards our application in conjunction with Audiveris; many common symbols that where not interesting in that context have been omitted, which severely limited the usability of *DeepScores* in other contexts. Second, *DeepScores* consist only of synthetically rendered music sheets, since labelling hundreds of thousands of music sheets by hand is prohibitively expensive. However, the common use case for OMR is scans or even photos of music sheets. This discrepancy can lead to severe performance drops between model training and actual use.

B. Enhanced character set

In an effort to make *DeepScores* more universally usable we created a new version—called *DeepScores-extended*—containing annotations for a far greater number of symbols. According to our knowledge and discussions with other members of the community, no crucial symbols are missing from the *DeepScores-extended* annotations. The full list of supported symbols is available online[1].

C. Richer musical information

While the interest of the authors lies in the detection of musical symbols, this task is not the full problem of OMR. The reconstruction of semantically valid music from detected symbols is at least as challenging as the detection. To enable research focused on reconstructing higher-level information, we have added additional information to the *DeepScores* annotations. Every labeled object now has an *onset* tag that tells the start beat of the the given object. All noteheads additionally have their relative position on the staff as well as their duration in their annotation (see Figure 1).

Fig. 1. Small piece of music notation with DeepScores-extended annotations overlayed. The naming is either classname.onset or classname.onset.relativecoordinate.duration, depending on availability.

D. Planned improvements

A drawback of the *DeepScores* dataset is that it is synthetic. We are currently working on a much smaller dataset, meant for transfer-learning, that consists of pages originally taken from *DeepScores* that are printed and then digitized again. Then, through a global centering and orientation alignment of the scan, the original annotations are made valid again for the scanned version. We use different printers, scanners, cell-phone cameras, and paper qualities to make the noise introduced by this process resemble the real world use case as much as possible. Naively training a Deep Watershed Detector on this new dataset, we observed that the detector was unable to find anything on the testing set despite that the loss function converged. This led us to believe that severe overfitting is going on, and we were able to get promising results by simply adding l2-regularization and performing more careful training (see Figure 2 for a qualitative result of the detector on the new dataset).

III. FURTHER RESEARCH ON DEEP WATERSHED DETECTION

A. Augmenting inputs

DeepScores, unlike many academic datasets, is extremely unbalanced. In fact, the most common class (notehead black)

*Equal contribution

[1] *tuggeluk.github.io/deepscores_syms_list*

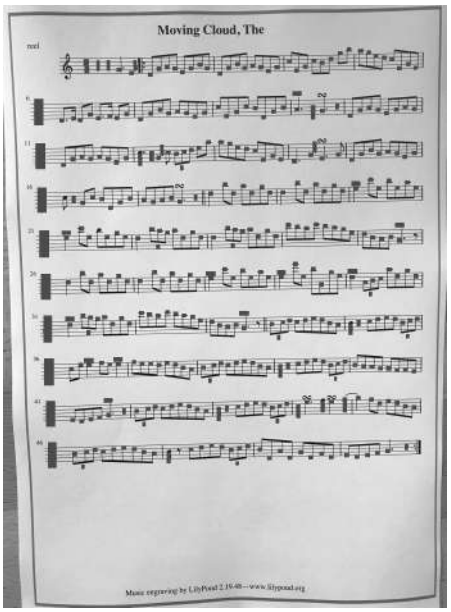


Fig. 2. Preliminary results of our model (grey boxes) on a photo of a printed sheet. While not perfect (for example, our model misses the clef in the first row), they already look promising.

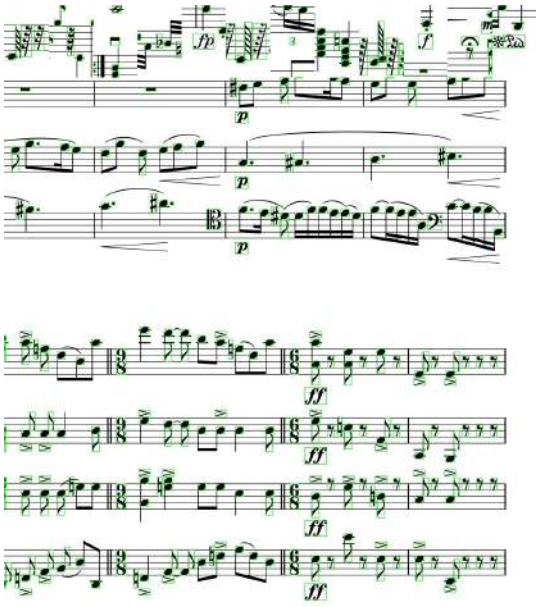


Fig. 3. A musical score where 12 small images have been augmented at the top of 7 regular staves. The bounding boxes are marked in green.

contains more symbols than the rest of the classes combined, while the top 10 classes contain more than 85% of the symbols. However, some of the rare symbols are important and simply dismissing them might lead to semantic problems during the reconstruction of valid music in some digital format. Initially, we tried to solve the problem by using a weighted loss function which penalizes more severely the mistakes on the rare symbols, but to no avail. In [3] we conjecture that the inbalance is so extreme that simply weighting the loss function leads to numerical instability, while at the same time the signal from these rare symbols is so sparse that it will get lost in the noise of stochastic gradient descent during the training: many symbols will be present only in a tiny fraction of mini batches. Both of these problems do not get solved by a weighted loss function.

Our current answer to this problem is oversampling rare classes by data synthesis, where we locate rare symbols in the dataset, and during training, we append these symbols at the top of the musical sheets (see Figure 3). More specifically, we augment each input page in a mini-batch with with 12 randomly selected synthesized crops of rare symbols (of size 130×80 pixels) by putting them in the margins at the top of the page. Directions on the choice of the creation of augmented symbols are given on [4]. This way, the neural network (on expectation) does not need to wait for more than 10 iterations to see every class which is present in the dataset. At the same time, we have been experimenting with pre-training the net with fully synthetic scores where the classes are fully balanced and then retraining it on the full *DeepScores* dataset. The two approaches are complementary and preliminary results show improvement, though more investigation is needed: overfitting on extremely rare symbols is still likely, and questions remain regarding how to integrate the concept of patches (in the margins) with the idea of a full page classifier that considers all context.

B. Cached bounding boxes

The biggest problem of the Deep Watershed Detector (DWD) on a fundamental level is that the bounding box regression is inaccurate. This is possibly due to the fact that convolutional networks produce smooth outputs, but the bounding box map can be very non-smooth. This "smoothing-bias" creates an averaging over all bounding boxes and leads to an overestimation of small bounding boxes and an underestimating of large ones. We currently address this issue by using cached bounding boxes per class as a prediction, being quite accurate for most classes but completely unusable for others. This is a not a satisfactory solution and has to be improved. We are considering multiple approaches including different bounding box encodings in the output layer or usage of the DWD localization as an object proposal system in an R-CNN style detection scheme.

References

[1] A. Rebelo, I. Fujinaga, F. Paszkiewicz, A. R. Marcal, C. Guedes, and J. S. Cardoso. Optical Music Recognition: State-of-the-Art and Open Issues. International Journal of Multimedia Information Retrieval, vol. 1, no. 3, pp. 173190, 2012.

[2] L. Tuggener, I. Elezi, J. Schmidhuber, M. Pelillo and T. Stadelmann. *DeepScores* - A Dataset for Segmentation, Detection and Classification of Tiny Objects. In Proceedings of the International Conference of Pattern Recognition, 2018.

[3] L. Tuggener, I. Elezi, J. Schmidhuber, and T. Stadelmann. Deep Watershed Detector for Music Object Recognition. In Proceedings of the 19th International Society for Music Information Retrieval Conference, 2018.

[4] T. Stadelmann, M. Amirian, I. Arabaci, M. Arnold, G. Duivesteijn, I. Elezi, M. Geiger, S. Lörwald, B. Meier, K. Rombach and L. Tuggener. Deep Learning in the Wild. In Proceedings of the 8th IAPR TC3 Workshop on Artificial Neural Networks in Pattern Recognition, 2018.

A Case for Intrinsic Evaluation of Optical Music Recognition

Jan Hajič jr.
Institute of Formal and Applied Linguistics
Charles University
Email: hajicj@ufal.mff.cuni.cz

***Abstract*—Evaluating Optical Music Recognition (OMR) has long been an acknowledged sore spot of the field. This short position paper attempts to bring some clarity to what are actually open problems in OMR evaluation: a closer look reveals that the main problem is finding an edit distance between some practical representations of music scores. While estimating these editing costs in the transcription use-case of OMR is difficult, I argue that the problems with modeling the subsequent editing workflow can be de-coupled from general OMR system development using an intrinsic evaluation approach, and sketch out how to do this.**

I. We Need a Music Score Edit Distance

Optical Music Recognition (OMR) has a known problem with evaluation [1]–[3]. We can approach OMR evaluation from two angles: extrinsic and intrinsic. By *extrinsic*, we mean evaluation in application contexts: how well does an OMR system address a specific need (such as retrieval, transcription, playback, ...)? *Intrinsic* evaluation asks a different question: how much of the information encoded by the music score has a given OMR system recovered? An example of extrinsic OMR evaluation can be found, e.g., in [4], where OMR is evaluated in the context of a cross-modal retrieval system; (partial) intrinsic evaluation is done i.a. in [5], where pitches and durations of recognized notes are counted against ground truth data. In this short position paper, I assess what the outstanding problems in evaluating OMR are, and propose intrinsic evaluation as a sensible way forward for OMR research.

The major problem in OMR evaluation is that given a ground truth encoding of a score and the output of a recognition system, there is no automatic method capable of reliably computing how well the recognition system performs that would (1) be rigorously described and evaluated, (2) have a public implementation, (3) give meaningful results. Other applications such as retrieval or extracing MIDI can be evaluated using more general methodologies. E.g., when using OMR to retrieve music scores, there is little domain-specific to defining success compared to retrieving other documents; any time MIDI output is required, metrics used to evaluate multi-f0 estimation can be adapted; score following has well-defined evaluation metrics at different levels of granularity as well. Within the traditional OMR pipeline [6], the partial steps (such as smbol detection) also can use more general evaluation metrics. However, when OMR is applied to re-typesetting music (which is arguably its original motivation), no evaluation metric is available.

In fact, computing an "edit distance" between a ground truth representation of a full music score and OMR output may be the only evaluation scenario where satisfactory measures are not available. The notion of "edit cost" [7] or "recognition gain" [8] that defines success in terms of how much time a human editor saves by using an OMR system is yet more problematic, as it depends on the specific toolchain used.

What can be done? One can try and implement such a metric. However, because cost-to-correct depends on the toolchains music editors use to work with OMR outputs, developing extrinsic evaluation metrics of OMR for transcription would require user studies at a scale which is not feasible for the few active OMR researchers. For these reasons, we argue it would be helpful for OMR development to have an *intrinsic* evaluation metric. After all, why address individual concerns that OMR users may have when full-pipeline OMR does have the potential to address *all* the application scenarios of OMR, as it attempts to extract *all* the information available from a music score?

II. Music Notation Formats are Problematic

A part of the edit distance problem lies in the ways music notation is stored digitally. MusicXML or MEI, which represent current best practices in open-source formats of digital representation of music scores, have some properties that make it difficult to compute a useful edit distance between two such files (useful in the sense that it would meausre either the amount of errors that an OMR system made, or the actual difficulty of changing one score to the other). Furthermore, the formats can encode the same score in multiple ways – e.g., MusicXML stores scores either measure-wise, or voice-wise.

Next, both formats are designed top-down, as trees that represent in their nodes both abstract concepts like a voice or note and graphical entities such as stems or beams. This implies that they cannot represent partial recognition results, and cannot encode syntactically incorrect notation. Furthermore, while the hierarchical structure mostly reflects the abstract structures of music such as voices and measures, it does not reflect the structure of music *notation*: local changes in the score can lead to several changes in the encoding that occur far apart, and vice versa. This is an inherent limitation of their tree structure.

The LilyPond format is impractical for anything but attempts at end-to-end OMR, as it hides much of the graphical representation in its engraving engine, and has so many ways of representing the same music that it is hard to meaningfully compare LilyPond files. The MuNG format [3] does to some extent overcome this locality problem by assuming a directed acyclic graph instead of a tree stucture, but it is limited to OMR ground truth and lacks conversions to other formats than MIDI.

The lesson here is that one should not bind intrinsic OMR evaluation to specific notation formats. After all, these formats change much faster than music notation itself. Rather, an evaluation metric should focus on inherent properties of music notation.

III. Arguing for Intrinsic Evaluation

Intrinsic evaluation of OMR systems means to answer the question *"How good is this system?"* without having to add, *"for this specific purpose?"* – thus de-coupling research of OMR methods from their individual use-cases, including the problematic score transcription. After all, music notation is the same regardless of whether it is being recognized for the purpose of searching a database or for producing a digital edition of the score.

There is no reason why this should not be possible: there is a finite amount of information that a music document carries, which can be exhaustively enumerated. It follows that we should be able to measure what proportion of this information our systems recover correctly. The benefit of intrinsic evaluation would be shedding the burden of accounting for score editing toolchains, independence on problematic music notation formats used in broad practice, and a clearly interpretable automatic metric for guiding OMR development (and potentially usable as a differentiable loss function for training full-pipeline end-to-end machine learning-based systems).

IV. A Roadmap

What would such an intrinsic evaluation metric measure? At the fullest, we expect two classes of outputs from an OMR system. First, a digital re-encoding of the score itself — creating a digital document that would convey exactly the same to a reader as the original. Second, recovering the semantic musical information: primarily the pitches, durations and onsets of notes (the minimum to build a MIDI representation of the given composition).

A thorough definition of error types in OMR was done by Bellini et al. [8]. They ask human evaluators to count errors for individual symbol types, and what they call "high-level" mistakes: pitch and duration attributes of note symbols. This seems like a good starting point from which to develop an automated intrinsic OMR evaluation metric.

The reason why [8] do not automate error-counting was a (then) lack of ground truth data. This has now been alleviated by the DeepScores dataset [9] at the low level, and MUSCIMA++ dataset [3] at both levels. The other step to automating the metric of [8] is aligning the recognition output and the ground truth score. At the graphical level, where the outputs are in principle symbol and their relationships, success can be measured using some graph similarity metric. At the semantic level, distance on lists of `(onset, duration, pitch)` triplets would be conditioned on some optimal alignment; DTW seems like a possible starting point for tractably finding this alignment, as it harshly penalizes ordering errors, which are rather critical due to the sequential nature of music. Given that noteheads can be thought of as carriers of the semantic information within the graphical level, the graph alignment function can also be used to directly find corresponding semantic triplets.

V. Finally

I hope this short paper will inspire discussion on the merits of intrinsic evaluation of OMR (I am especially keen to find out how I am wrong!), and perhaps nudge along the musical score edit distance problem that has been a thorn in the side of OMR research for the duration of its existence.

Acknowledgments

This work is supported by the Czech Science Foundation, grant P103/12/G084, the Charles University Grant Agency, grants 1444217 and 170217, and by SVV project 260 453.

References

[1] M. Szwoch, "Using MusicXML to Evaluate Accuracy of OMR Systems," *Proceedings of the 5th International Conference on Diagrammatic Representation and Inference*, pp. 419–422, 2008. [Online]. Available: http://dx.doi.org/10.1007/978-3-540-87730-1_53

[2] Donald Byrd and Jakob Grue Simonsen, "Towards a Standard Testbed for Optical Music Recognition: Definitions, Metrics, and Page Images," *Journal of New Music Research*, vol. 44, no. 3, pp. 169–195, 2015. [Online]. Available: http://dx.doi.org/10.1080/09298215.2015.1045424

[3] J. Hajič jr. and P. Pecina, "The MUSCIMA++ Dataset for Handwritten Optical Music Recognition," in *14th International Conference on Document Analysis and Recognition, ICDAR 2017, Kyoto, Japan, November 13 - 15, 2017*, Dept. of Computer Science and Intelligent Systems, Graduate School of Engineering, Osaka Prefecture University. New York, USA: IEEE Computer Society, 2017, pp. 39–46.

[4] S. Balke, S. P. Achankunju, and M. Müller, "Matching Musical Themes based on noisy OCR and OMR input," pp. 703–707, 2015.

[5] Victor Padilla, Alan Marsden, Alex McLean, and Kia Ng, "Improving OMR for Digital Music Libraries with Multiple Recognisers and Multiple Sources," *Proceedings of the 1st International Workshop on Digital Libraries for Musicology - DLfM '14*, pp. 1–8, 2014. [Online]. Available: http://dx.doi.org/10.1145/2660168.2660175

[6] Ana Rebelo, Ichiro Fujinaga, Filipe Paszkiewicz, Andre R. S. Marcal, Carlos Guedes, and Jaime S. Cardoso, "Optical Music Recognition: State-of-the-Art and Open Issues," *Int J Multimed Info Retr*, vol. 1, no. 3, pp. 173–190, Mar 2012. [Online]. Available: http://dx.doi.org/10.1007/s13735-012-0004-6

[7] J. Hajič jr., J. Novotný, P. Pecina, and J. Pokorný, "Further Steps towards a Standard Testbed for Optical Music Recognition," in *Proceedings of the 17th International Society for Music Information Retrieval Conference*, M. Mandel, J. Devaney, D. Turnbull, and G. Tzanetakis, Eds., New York University. New York, USA: New York University, 2016, pp. 157–163. [Online]. Available: https://18798-presscdn-pagely.netdna-ssl.com/ismir2016/wp-content/uploads/sites/2294/2016/07/289_Paper.pdf

[8] Pierfrancesco Bellini, Ivan Bruno, and Paolo Nesi, "Assessing Optical Music Recognition Tools," *Computer Music Journal*, vol. 31, no. 1, pp. 68–93, Mar 2007. [Online]. Available: http://dx.doi.org/10.1162/comj.2007.31.1.68

[9] Lukas Tuggener, Ismail Elezi, Jürgen Schmidhuber, Marcello Pelillo, and Thilo Stadelmann, "DeepScores - A Dataset for Segmentation, Detection and Classification of Tiny Objects," *CoRR*, vol. abs/1804.00525, 2018. [Online]. Available: http://arxiv.org/abs/1804.00525

HISPAMUS: Handwritten Spanish Music Heritage Preservation by Automatic Transcription

José M. Iñesta, Pedro J. Ponce de León, David Rizo, José Oncina, Luisa Micó,
Juan Ramón Rico-Juan, Carlos Pérez-Sancho, Antonio Pertusa
Software and Computing Systems, University of Alicante, Spain
{inesta,pierre,drizo}@dlsi.ua.es

***Abstract*—The HISPAMUS proposal aims at enhancing the Hispanic music heritage from the 15th to the 19th centuries, by exploiting the digital resources of these collections. In addition, thousands of oral tradition melodies that were compiled by folklorists in the 1950s decade are digitized just as images, currently without the possibility of content-based search or study. It is necessary to develop services and tools for the benefit of archives, libraries, scholars, computer scientists and general public. HISPAMUS tries to provide smart access to archival manuscripts of music scores, allowing its reuse and exploitation. In order to reach this ambitious goal, our group can provide cutting-edge technology in the fields of Machine Learning, Pattern Recognition, and Optical Music Recognition.**

***Index Terms*—Heritage, Notation transcription, Encoding, Optical music recognition, Handwritten, Mensural notation.**

I. Introduction

Musical notation has evolved over the centuries into various writing systems. In particular, Spanish white mensural notation was the dominant code for writing music in Spain (and in Latin America) between the 16th and 18th centuries, producing large collections of handwritten documents yet to be made accessible to the public.

All this cultural heritage has been, in part, hidden from the public eye because its custody is restricted to certain actors (the Church, private collections, etc.) and its interpretation by contemporary musicians is far from their correct understanding. Accordingly, musicologists work for the study and appreciation of such compositions, from the cultural and economic point of view, is limited by the tools within their reach. In all these cases, just the scanned or photographed images of the scores are available, and not their transcription into symbolic music formats. How do we make them digitally available to the public for its use and valorization?

The HISPAMUS project aims at generating the software tools needed to convert manuscript score images to a modern digital format, like MEI/MusicXML, ready to render modern notation scores, allowing the public to enjoy and search into the digital contents of these works, either as a musicologist or as a musician.

The main objectives of the project can be summarized as follows:

- Foster a significant technology progress beyond the current state of the art in digital music content production and management by means of optical music recognition and machine learning technologies.
- To make it possible to access both early and traditional music works in new ways, for study, analysis, performance, etc., within the framework of new technologies and web services.

The OMR techniques available today are not ready for being applied to these kind of handwritten documents. Thus, one of the main goals of this project is to process this kind of documents, covering the whole workflow from the digitized images to the production of the digital symbolic format.

One important feature will be the possibility to render also a translated version to a modern notation score, readable by a contemporary musician, and browse the digital contents of these works, either as a musicologist or as a musician.

From the technical point of view, the processing stages (see Fig. 1) will be designed using machine learning (ML) techniques. Most available systems are based on heuristic rule methods, but one of the problems of OMR is the existence of too many repertoire-dependent context rules [1]. Therefore, it is hard to extend them to recognize early music notations. By applying ML technologies we can build models based on *training pairs*: sets of data examples presented together with a ground-truth label. This opens up the possibility of adapting the system, in principle, to any music notation, if labeled data is available for the system to learn.

II. Methodology

The software to be developed through the HISPAMUS project is required to recognize the contents of printed or manuscript scores from different notations, the encoding into all current standards, and the assisted transcription into a edited version, suitable for preparing a critical edition. In addition, as this system is conceived as a research tool, it has to be equipped with features to measure the efficiency and effectiveness of the different tasks and approaches.

The scheme in Fig. 1 describes the main workflow in the project. The system is fed with digitized images of the manuscripts. The user has to organize the images according to the books. The OMR processes include a layout analysis for separating music from texts, and identifying the different regions in the pages.

In all cases, our OMR operations generate two kinds of sequence symbol encodings. One is named *agnostic*, which recognize symbols by their shape and position, without analyzing the musical meaning of the symbol in the score. The

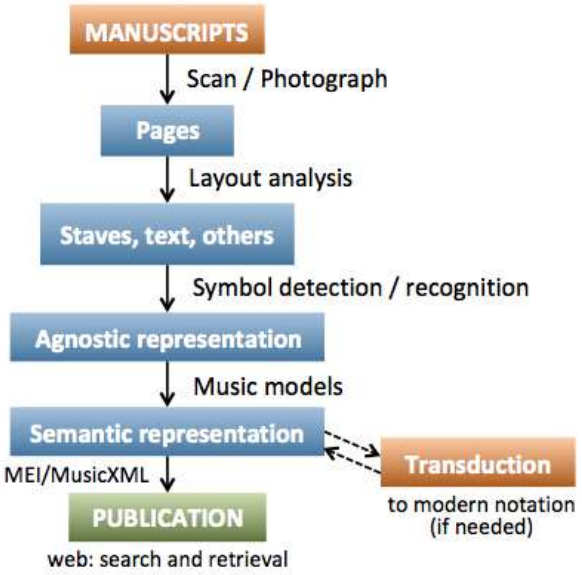


Fig. 1. Main workflow in HISPAMUS.

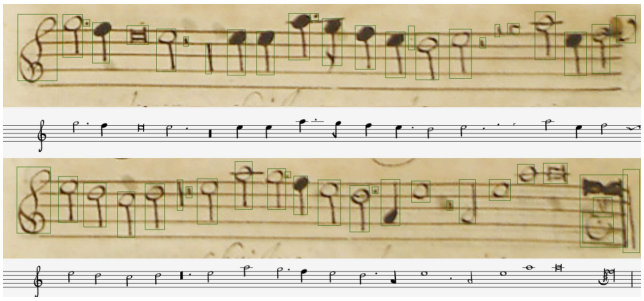


Fig. 2. Snapshot of the system in its current stage of development. Bounding boxes for the symbols and their recognition are displayed.

other encoding is the *semantic* one, where meaningful musical information is encoded into a standard music format.

The system is designed under the assumption that no OMR approach is able to provide a perfect performance. Thus, the user intervention is required. This assessment and correction task can be accomplished at graphical symbol level (agnostic) or by editing the music content obtained directly (semantic).

From our experience, we believe that it is difficult to find a single approach for OMR that can be effective in all situations. Therefore, we implement three types of automatic recognition, analyzing the image at different levels of granularity:

a) User-driven symbol recognition: the first approach is based on the work of Calvo-Zaragoza et al. [2], where the user locates the symbols manually. Typically, the computer mouse can be used to indicate the position of the symbols, but we also allow in MuRET, the software application developed for the project, the use of a digital pen, resulting in a more ergonomic interaction.

b) Holistic staff recognition: in this approach, each staff is processed without previous symbol segmentation. This can be achieved by using continuous models like hidden Markov models [3] or recurrent neural networks [4].

c) Full page recognition: the last approach we consider is a neural model that is applied directly to a full page [5]. In this case, given the image, the model returns a list of predictions, each of which indicates a bounding box and the category of the object therein.

These approaches work at the graphical level (agnostic representation), ignoring the semantics of the music notation itself that can be recovered in a post-processing stage. For example, in the case of notes, the vertical position within the staff, along with the clef and eventual alterations, allows inferring their pitches.

Our intention is also to take advantage of this infrastructure to carry out user-centered studies, which will provide information about which of these approaches is the most effective in each situation. The ultimate goal is to obtain the structured encoding of the music sources with the least possible effort from the user. The effort of the user is closely related to the accuracy of the recognition models, but also with the type of corrections and interactions required.

Music editing: As mentioned above, one of the fundamental objectives of HISPAMUS is to obtain a transcription of the manuscript without any ambiguities, so that both the musicologist and the publisher can generate a critical edition from it. This stage includes two tasks: the optional correction of the recognized graphical symbols from the OMR, and the correct assignment of musical functions to those graphical symbols.

Furthermore, the current goal of MuRET is not to generate final edited preprints, but to produce contents to be sent to on line services or publishers. As mentioned above, the system should export in any interchange format will be edited by these publishers to fulfil their publishing workflow. Thus, a printed image must be generated with the aspect of how approximately the transcription should look like. MuRET exports this kind of output using PDF.

Acknowledgements

HISPAMUS is the project TIN2017-86576-R, supported by the Spanish Ministry, partially funded by the EU.

References

[1] D. Byrd and J. Simonsen, "Towards a standard testbed for optical music recognition: Definitions, metrics, and page images," *Journal of New Music Research*, vol. 44, no. 3, pp. 169–195, 2015.

[2] J. Calvo-Zaragoza, D. Rizo, and J. M. Iñesta, "Two (note) heads are better than one: Pen-based multimodal interaction with music scores," in *17th International Society for Music Information Retrieval Conference*, 2016, pp. 509–514.

[3] J. Calvo-Zaragoza, A. H. Toselli, and E. Vidal, "Handwritten music recognition for mensural notation: Formulation, data and baseline results," in *14th IAPR International Conference on Document Analysis and Recognition*, 2017, pp. 1081–1086.

[4] J. Calvo-Zaragoza and D. Rizo, "End-to-end neural optical music recognition of monophonic scores," *Applied Sciences*, vol. 8, no. 4, pp. 606–629, 2018.

[5] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 580–587.

Advancing OMR as a Community: Best Practices for Reproducible Research

Alexander Pacha
Institute of Visual Computing and Human-Centered Technology
TU Wien
Vienna, Austria
alexander.pacha@tuwien.ac.at

***Abstract*—Optical Music Recognition has been under investigation for over 60 years but remains an unsolved problem, because research happens distributedly, often without reusability in mind. As scientists, it should be one of the goals to share knowledge in a way, that it becomes accessible and useful for someone else to build on top. Without that, one's effort is often doomed to rot in a drawer. To oppose this development, not only the paper, but also the source code, datasets, and executables should be made publicly available for the community to finally advance beyond the state, where the wheel is reinvented every time a new researcher joins the field.**

***Index Terms*—optical music recognition, reproducible research**

I. Introduction

Optical Music Recognition (OMR) has seen decades of research with many questions remaining unsolved [1]. Solutions, algorithms, and systems were frequently developed independently, trying to solve particular (sub-)problems [2]–[7]. However, without the perspective of what happens to a project after the scientific publication has been accepted, the advances we have achieved remain scattered across the world. It would be great to share datasets, algorithms, results, and knowledge alike. This is slowly becoming the scientific norm, but to leverage the work of someone else, the author has to do more than just providing a link to the source-code: properly documenting the code base, providing automatic tests that can be executed on a continuous integration platform and ideally one-click solutions for running the application are necessary. These things are part of the standard repertoire used in industry projects and should be equally relevant in scientific projects from the field of computer science.

II. Publish reproducible results

There are three kinds of artifacts that researchers commonly share with the community: knowledge, primarily in the form of scientific articles, datasets, and source code. Papers usually undergo a scientific peer-review to ensure a certain quality level and are published in journals, or presented at conferences. Many of these publications now come with links to the source code that is hosted on Github or similar platforms [8]–[12]. Sharing access to the repository allows not only to see the final result, but also the steps that led to that result. However, the other two artifacts often lack a rigorous review or are not review at all. When reviewing another peer's work, demand the code as supplementary material to verify reproducibility. If understanding their code turns out to be too hard for another expert working in the same field, it is unlikely that anyone will use the reviewed work. If so, let the authors know by adding an appropriate remark in the review.

A. Source Code

When publishing source code, it should contain at least a README file, which summarizes the purpose of this repository as well as how to build and run the application. Keep in mind, that good names for methods, variables, and packages are preferred to extensive commenting in the code [13]. Sensible documentation that is generated directly from the source code[1] can help other users to quickly get started, ideally with a few examples on how to use your project. http://muscima.readthedocs.io/en/latest/Tutorial.html is an example of a good tutorial. To further promote free reuse, an appropriate license should be attached to the repository, e.g., the MIT license. https://tldrlegal.com/ contains a concise summary of many popular licenses. Note, that if no license is specified, default copyright applies that restricts the allowed usage heavily.

B. Executables

Use a continuous integration platform, such as Travis[2], to prevent unexpected regressions in the code. Many of these services are freely available for open-source projects. Some of them can also be used as a platform to provide executables for potential users. If an application has a complex environment and requires specific dependencies, consider using containerization, e.g., with Docker[3]. The additional benefit of setting up an application as a ready-to-use container is a possible integration with environments like the DIVAServices [14], that provide interactive demos as part of their website[4]. Such an interactive demo lets other researchers explore your work without the huge upfront effort, that is sometimes required to get an application to run locally.

[1]https://readthedocs.org/
[2]https://travis-ci.org/
[3]https://www.docker.com/
[4]http://divaservices.unifr.ch/spotlight

C. Datasets

A delicate matter concerns the creation of datasets. This sore point is still a major issue within the OMR community. Although there are a few large datasets that have recently been published or that will be published in the near future, these datasets are often not compatible with each other due to different notation formats, encodings, and granularity of the underlying information. WoRMS offers a great possibility to address this issue as a community. From a practical point of view, datasets will probably always be encoded somewhat differently, but if the authors claim, that their dataset can (in theory) be exported into a particular format, it is their responsibility to provide a respective converter. As a general guideline, prefer a simple, plain format (e.g., comma-separated-values) that can easily be converted into other data formats. It is also beneficial, if additional visualization functions are provided, that can help to inspect intermediate and display final results. Finally, to gain visibility, a dataset can be listed on the website of the OMR datasets project (http://apacha.github.io/OMR-Datasets/) by simply creating a pull-request or adding an issue.

D. Showcases

Two projects demonstrate the implementation of above-mentioned guidelines: The OMR datasets project[5] and the Music Object Detector[6] [15]. Their source code is freely available under an MIT license, and detailed README files explain how to use the source code, how to reproduce the results and how to test the final system with pre-trained models and demo-scripts. The Music Object Detector is also available via the DIVAServices Spotlight, where arbitrary music scores can be uploaded to test the object detection capabilities easily.

III. Do not reinvent the wheel

OMR has seen many attempts to "solve it" but it is still considered an unsolved problem for many scenarios. Often, users are disappointed when they try OMR on their dataset and see themselves confronted with its insufficiencies. This is caused by applications not being designed to handle arbitrary scores: hard-coded variables, assumptions on the layout or the limitation to a particular music notation system lead to programs that only work well for the dataset for which they were developed. This pitfall can be avoided by letting actual users test your system and give you honest feedback. When reflecting on the achievements as a community, many things were attempted but failed so far. Agreeing on an output format or a standard vocabulary are good examples. Another one is how to evaluate entire OMR systems. Despite many proposed metrics, you simply cannot compare two systems that were designed with entirely different purposes in mind. Scientific competitions can help to mitigate this problem because the task needs to be well defined, an appropriate dataset has to be provided, and all participants must follow the same evaluation protocol. The 2012 Music Scores Competition [16] is an example that led to a range of robust and generalizable solutions for staff removal. And even many years after the competition, the dataset is still being used intensively for a wide range of tasks, thanks to its liberal license. Finally, it has to be said, that we should avoid reinventing the wheel over and over again and not only make sure that our research is usable by other researchers, but also that we actively leverage the work of others and build on top of their work, instead of always starting over from scratch.

[5] http://apacha.github.io/OMR-Datasets/

[6] http://github.com/apacha/MusicObjectDetector-TF

References

[1] A. Rebelo, I. Fujinaga, F. Paszkiewicz, A. R. Marcal, C. Guedes, and J. S. Cardoso, "Optical music recognition: state-of-the-art and open issues," *International Journal of Multimedia Information Retrieval*, vol. 1, no. 3, pp. 173–190, 2012.

[2] J. Ashley, B. Laurent, P. Greg, and E. I. Fujinaga, "A comparative survey of image binarisation algorithms for optical recognition on degraded musical sources," *ACM SIGSOFT Software Engineering Notes*, vol. 24, no. 1985, pp. 1994–1997, 2008.

[3] C. Dalitz, M. Droettboom, B. Pranzas, and I. Fujinaga, "A Comparative Study of Staff Removal Algorithms," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 30, no. 5, pp. 753–766, 2008.

[4] A. Fornés, "Primitive Segmentation in Old Handwritten Music Scores," in *International Workshop on Graphics Recognition 2005*, 2005, pp. 279–290.

[5] P. Bellini, I. Bruno, and P. Nesi, "An off-line optical music sheet recognition," in *Visual Perception of Music Notation: On-Line and Off Line Recognition*. IGI Global, 2004, pp. 40–77.

[6] D. Blostein and H. S. Baird, *A Critical Survey of Music Image Analysis*. Berlin, Heidelberg: Springer Berlin Heidelberg, 1992, pp. 405–434.

[7] J. Calvo-Zaragoza, A. H. Toselli, and E. Vidal, "Early handwritten music recognition with hidden Markov models," in *2016 15th International Conference on Frontiers in Handwriting Recognition (ICFHR)*. Institute of Electrical and Electronics Engineers Inc., 2017, pp. 319–324.

[8] A. Pacha and H. Eidenberger, "Towards a universal music symbol classifier," in *Proceedings of the 12th IAPR International Workshop on Graphics Recognition*, New York, USA, 2017, pp. 35–36.

[9] J. Calvo-Zaragoza and D. Rizo, "End-to-end neural optical music recognition of monophonic scores," *Applied Sciences*, no. 4, 2018.

[10] L. Tuggener, I. Elezi, J. Schmidhuber, and T. Stadelmann, "Deep watershed detector for music object recognition," in *Proceedings of the 19th International Society for Music Information Retrieval Conference*, 2018.

[11] J. Hajič jr. and M. Dorfer, "Prototyping full-pipeline optical music recognition with muscimarker," in *Extended abstracts for the Late-Breaking Demo Session of the 18th International Society for Music Information Retrieval Conference*, 2017.

[12] K. MacMillan, M. Droettboom, and I. Fujinaga, "Gamera: Optical music recognition in a new shell," *Proceedings of the International Computer Music Conference*, pp. 1–4, 2002.

[13] A. Shvets, G. Frey, and P. Marina, "Source making." [Online]. Available: http://sourcemaking.com/refactoring/smells/comments

[14] M. Würsch, M. Liwicki, and R. Ingold, "Web Services in Document Image Analysis - Recent Developments and the Importance of Building an Ecosystem," in *13th IAPR Workshop on Document Analysis Systems*, Vienna, Austria, 2018, pp. 334–339.

[15] A. Pacha, K.-Y. Choi, B. Coüasnon, Y. Ricquebourg, R. Zanibbi, and H. Eidenberger, "Handwritten music object detection: Open issues and baseline results," in *2018 13th IAPR Workshop on Document Analysis Systems (DAS)*, 2018, pp. 163–168.

[16] A. Fornés, A. Dutta, A. Gordo, and J. Lladós, *The 2012 Music Scores Competitions: Staff Removal and Writer Identification*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2013, pp. 173–186.

Digitisation and Digital Library Presentation System – Sheet Music to the Mix

Tuula Pääkkönen
National Library of Finland
University of Helsinki
Mikkeli, Finland
tuula.paakkonen@helsinki.fi

Jukka Kervinen
National Library of Finland
University of Helsinki
Mikkeli, Finland
jukka.kervinen@helsinki.fi

Kimmo Kettunen
National Library of Finland
University of Helsinki
Mikkeli, Finland
kimmo.kettunen@helsinki.fi

***Abstract*—The National Library of Finland (NLF) has done long-term work to digitise and make available our unique collections. The digitisation policy defines what is to be digitised, and it aims not only to target both rare and unique materials but also to create a large corpus of certain material types. However, as digitisation resources are scarce, the digitisation priorisation is done annually. This involves the library juggling the individual researcher needs with its own legal preservation and availability goals. The digital presentation system at digi.nationallibrary.fi enables fast operation by being close by to the digitisation process to enable a streamlined flow of material from production to the end users. In this paper, we will describe our digitisation process and its cost-effective improvements, which have been recently applied at the NLF. In addition, we present the enrichment phase of digitisation, which could be applicable with processing of musical sheets.**

***Index Terms*—digitisation process, digital presentation system, research and development projects, digital chain**

I. Introduction

As part of its preservation and accessibility goals, the National Library of Finland (NLF) has been digitising its collections from the year 1998, when the Centre for Digitisation and Preservation was created as a unit to Mikkeli. Since then, based on the resources available, approximately 1 million pages have been digitised annually and put into a presentation system for the public to view. At the moment, for example, the number of digitised newspapers and journals has recently exceeded 13 million pages [1], which can be viewed at the digi.nationallibrary.fi In addition, the doria.fi service contains books, maps and images and audio.

The digitisation process, which has been formed in the National Library, has defined the key steps that are crucial for a successful digitisation workflow. The library users usually associate digitisation with only the scanning phase; however, own expertise is also needed before and after scanning to ensure a smooth digitisation workflow. In our environment, we have defined a model called a digitisation chain, which describes the tasks of digitisation process.

Part of this work is funded by the Academy of Finland project COMHIS Computational History and the Transformation of Public Discourse in Finland, 16401910, decision number 293341.

II. Digitisation – what and how?

The digitisation policy of the NLF was created in 2010 [2]. As stated in the policy, accessibility, preservation and ongoing use are the aims of digitisation. In essence, the idea is to preserve the unique collections this is the reason, for example, that the most-used materials are to be digitised so that they are more easily accessible to those who need them. In the long run, this also minimizes the efforts of the customer-service personnel, since the physical materials do not need to be brought to the researchers desks.

A. The Renewed Digital Chain

The duty of the NLF is to deposit and preserve everything published in Finland. According to the Legal Deposit Act, the NLF receives a copy of each newspaper and magazine published in Finland. Received publication materials are processed in the library according to an internal concept called the digital chain. Processing of the publication material in the digital chain consists of the following five phases: 1) material deposit and return (including cataloguing); 2) preparation, scanning and conservation (if needed); 3) post-processing, which includes structural analyses; 4) microfilming from digital version; and 5) deployment, use and preservation. The digital chain is schematically presented in Fig. 1.

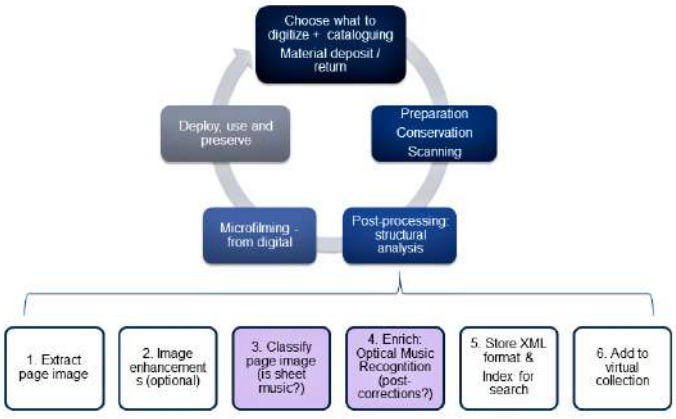


Fig. 1. The digital chain with enrichment tasks of NLF.

Processing of sheet music is a case example of the enrichment phase, which is starting to form within the post-processing phase. After we get the page image, we can classify it and based on that result take specific enrichment path. In case of sheet music, doing optical music recognition (OMR) with external tools and storing its results in standardized format. Finally indexing and automatically adding material to correct virtual collection enables end-users to find material.

B. Increasing Amount of Material Types

The realm of publication is constantly changing, which requires changes also to the digital library presentation system, by providing support to new material types. Currently we have a project for adding support to monographs (books, catalogues, and possibly maps) to the digital presentation system and music sheets could be possible next step.

There is increasing interest towards various music materials which are digitized within the NLF, so highlighting them at the same presentation system would ease processing requirements. This could include processing of the notes and scores, utilizing their metadata and processing them in a user-friendly way, i.e. by providing a separate search for musical notes. At the moment, the musical notes are recognized in the post-processing of the digitisation as illustrations. There has been initial experiments to classify illustrations via TensorFlow library [3], which in first step requires creating a training set of sheet music illustrations in order to find more of them. Now the search of the illustrations happens via the recognized text on the page. In future it would be useful to add tags to the illustrations and to get to more granular level of utilizing illustrations, and music sheets. Fig. 2 shows how the presentation system shows a page image with sheet of music on the right functions bar, then page image and on the left the recognized text of the page.

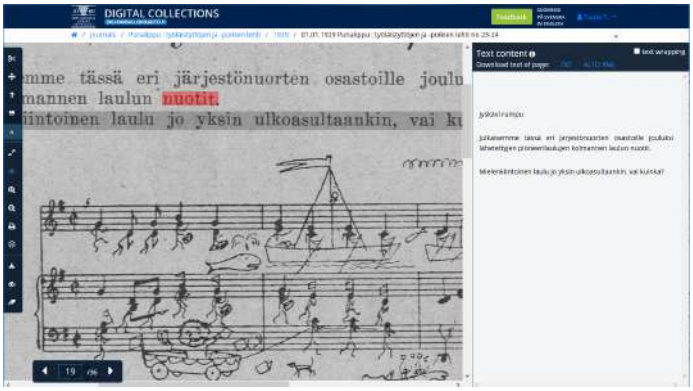


Fig. 2. Sheet of music in the presentation system. [4]

C. Planning with Resource Constraints in Mind

Adding musical sheets and notes provide intriguing challenges for the digitisation process and to the requirements of the presentation system. As with any material creating metadata of different music document types requires work with cataloguing. E.g. Bavarian state library lists monographs, musicological journals, sheet music, music manuscripts, sound carriers and libretti [5]. More technical issue is how should the post-processing of the digitisation identify and utilize the musical notes, which OMR-quality level would be acceptable? Do stakeholders want OMR? How results from OMR could be utilized in the presentation system? In addition, there are multiple formats, protocols and tools to choose from [6] where selection requires evaluation. For example, the International Image Interoperability Framework (IIIF) for illustrations could be useful for the music sheets images, too. The annotation support of the IIIF standard could enable crowdsourced fixes to the most complicated OMR cases, like we have previously already seen in the text recognition research [7]. Most effort will be put to the configuration of digitisation process to be able to process different kinds of sheet music documents. For manual annotation or quality control we could only put a few days of work. Therefore, in the enrichment phase existing open-source tools could be invaluable when they could be plugged-in to the digitisation process with minimum changes.

III. Conclusions

The digitised collections of a library create a new kind of environment in which existing data are used in different ways [8]. This creates new challenges with regard to the copyrights, use, re-use and availability of the materials. The additional enrichment of the materials could also be done together with the researchers. In that case the library would act as content provider but also utilize the research in the presentation system. For example, in Denmark, sheet music is in the top-5 search words [9], so improving access could hopefully increase awareness of that collection. Any OMR technology chosen should fit to the existing digital chain, and provide standardized format for preservation and further use.

In the long run, in addition to the digital chain, a new enrichment chain can be created to include additional automatic enrichments, which aim to ease the researcher use of the materials. Sheet music could act as one more material type, which could invite more users to the collections.

References

[1] H. Arpiainen, "Digiss jo yli 13 miljoonaa sivua aineistoja," Mar 2018, (in Finnish). [Online]. Available: https://tinyurl.com/digi1813

[2] N. L. of Finland, "The digitisation policy of the national library of finland," 2010. [Online]. Available: http://urn.fi/URN:NBN:fi-fe201401151119

[3] TensorFlow, "Tensorflow for poets." [Online]. Available: https://tinyurl.com/digicl

[4] *Punalippu: tylistyttjen ja -poikien lehti*, no. 2324, p. 19, Jan 1929.

[5] J. Diet, "Digitization and presentation of music documents in the bavarian state library," *Fontes Artis Musicae*, vol. 61, no. 3, p. 275283, Jul 2014.

[6] R. Freedman, R. Viglianti, and A. Crandell, "The collaborative musical text," *Music Reference Services Quarterly*, vol. 20, no. 34, p. 151167, Oct 2017.

[7] K. Kettunen and T. Pääkkönen, "Measuring lexical quality of a historical finnish newspaper collection analysis of garbled ocr data with basic language technology tools and means," May 2016, p. 956961. [Online]. Available: http://www.lrec-conf.org/proceedings/lrec2016/index.html

[8] KB.se, "Digitaliserade svenska dagstidningar 1758-1926 (sida 1) av kb digitaliserade dagstidningar, andra tjnster," 2017, (in Swedish). [Online]. Available: http://feedback.tidningar.kb.se/viewtopic.php?id=89

[9] A. . Jensen, "Experiencing materials and knowledge: The danish national librarys music collection," *Fontes2011*, vol. 58, no. 3, p. 253259, Jul 2011.

Music Search Engine from Noisy OMR Data

Sanu Pulimootil Achankunju
Bavarian State Library
Munich, Germany
sanu.pulimootil-achankunju@bsb-muenchen.de

***Abstract*—The music department of the Bavarian State Library (BSB) is one of the internationally leading music libraries. A huge collection of our music scores are only accessible via meta data based search. So we have started a new Music Information Retrieval (MIR) project to enable content wise searching in printed music scores. To this end, we extracted the symbolic note information from the entire works of four famous composers using Optical Music Recognition (OMR) technology, which transforms sheet music or printed scores into a machine readable format. The OMR data are quite noisy containing numerous extraction errors. We have created a music search engine to enable melody search on this noisy data, that can still achieve a very good retrieval quality. We also report on our experiments to test the quality of this search engine with musical themes from an external source.**

***Index Terms*—Optical Music Recognition, Music Information Retrieval, melody search, N-gram**

I. Introduction

Full-text search is now a common concept in many digital libraries due to the advancements in Optical Character Recognition technology. However, that is not the case when dealing with musical documents because OMR technology is far from being perfect [1] [2]. Meta-data based search is the most prominent way of accessing these documents. In order to enhance the search procedure, BSB has started a new project to enable content wise searching in the printed music scores. In the first phase of this project, many OMR software applications were subjectively tested on a small subset of digitized music sheets to find out the best software that works well with our images. In the second phase, sheet music images were converted into symbolic music using the selected software. In the third phase, a web application named "musiconn scoreseach" was developed to enable melody search on this data. Later phases of this project will support features like theme recognition/search, polyphony, rhythm search, etc. In this paper, we deal with a MIR scenario where noisy OMR data form the base of a music search engine.

II. Processing Pipeline

After comparing SharpEye, Audiveris, SmartScore X^2 and Capella Scan, we have decided to use SmartScore X^2 Professional v 10.5.8 for our project as the OMR software. This software was installed on a dedicated server and the sheet music images are sent to it automatically for processing. Complete editions of the works of Ludwig van Beethoven, Georg Friedrich Händel, Franz Schubert and Franz Liszt were converted into symbolic music using this software. As a result, about 40,000 MusicXML files were created from about 44,000 images including non-music sheets like content, introduction and blank pages. Due to the large quantity, manual correction of the noisy OMR data is a difficult task. Each parts of the partwise MusicXML file are processed separately. Polyphony is removed to reduce the complexity of index and search process. Only pitch sequences are considered for the melody search engine. Monophonic pitch sequence is created by considering only the highest value of pitch at a given instance of time, i.e., we take the upper pitch contour of the given polyphonic part. If the part has more than one staff, we extract the monophonic pitch sequence of each individual staff and the same of the given part considering all the staves as a single piece.

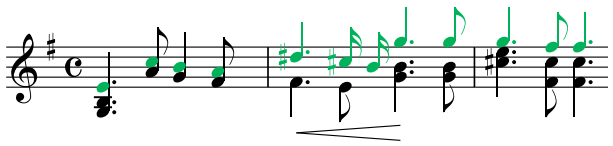


Fig. 1. Sample OMR output from Händel's "Der Messias: Oratorium"

Let us consider the sample OMR output given in Fig. 1. Each element of the pitch sequence consists of the note name and its octave number. Monophonic pitch sequence generated from the OMR data is given as `E4 C5 B4 A4 D♯5 C♯5 B4 G5 G5 G5 F♯5 F♯5` and is highlighted in the figure. In addition to the pitch sequences, we convert each of these also into sequences of intervals. Interval sequence is nothing but the difference of adjacent pitches in the sequence. An advantage of this sequence is its invariance to transposition [3]. OMR procedure can misinterpret the global information such as clef [4] or might detect wrong notes. The interval sequence helps us to get the melodic contour. Interval sequence for the above given example is `8, -1, -2, 6, -2, -2, 8, 0, 0, -1, 0`.

After extracting pitch and interval sequences from the parts of a MusicXML file, they are sent along with their metadata to our search engine for indexing. From these sequences, N-grams of different sizes are indexed in the search engine. N-grams can be considered as a sequential list of n words. The task of our search engine is to find out the documents that have the same N-grams as the query.

This work has been supported by the German Research Foundation (Deutsche Forschungsgemeinschaft) in the grant "Fachinformationsdienst Musikwissenschaft - 1. Fortsetzungsantrag", CE 134/27-2, Jan 17 - Dec 19.

III. Search Engine Interface

We have also created a user friendly front end for the search engine. A screenshot of this web interface is shown in Fig. 2. Query input to the search interface can be given using a virtual piano. A minimum of three notes should be given as a query and it can be any notes from `C2` to `C7`. As a visual feedback,

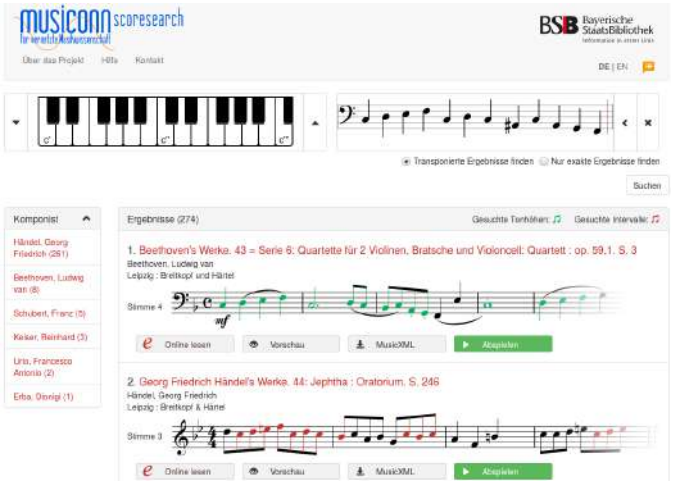


Fig. 2. Screenshot of the Music Search Engine

the notes given through the piano will appear on the editable musical staff. Notes can also be given through the keyboard shortcut keys and it will be inserted into the current cursor position that can be changed by clicking the staff. An audio feedback is also given as each note is pressed. Higher or lower octaves can be selected using buttons to the right and left of the piano. The default option will search the absolute pitches and intervals together. There is also an option to search only the pitches. A search on an average takes only about 4-5 seconds.

The search results will appear with its metadata. Search result displays some measures where the notes are found. Pitches or intervals detected in these measures are highlighted in different colours as shown in Fig. 2. There are options to open the exact page of the original document, to see the whole page rendered from MusicXML data with all the search results highlighted, to download the particular MusicXML file and to listen to the search results. Faceted navigation is also enabled to refine the search results.

IV. Retrieval Experiment and Result

In order to evaluate the retrieval quality, we use MIDI files from the "Electronic Dictionary of Musical Themes"[1]. It contains roughly about 10000 monophonic musical themes of instrumental western classical music. For our evaluation purposes, we considered only about 1200 of them from the works of same composers mentioned in II. Using the metadata, ground truth information were made for 200 random files and pitch data were extracted from these files. Random queries were made from these data and were sent to the search engine. Each query contains about 6 to 12 notes, not necessarily from the beginning of a theme. In a normal search engine scenario, a user is only interested in the first few matches. So in our experiment, for each query we consider the top K matches or the first K results of the search engine for some number $K \in \mathbb{N}$. A search is considered successful if the top K matches contain the relevant document. In our retrieval process, we

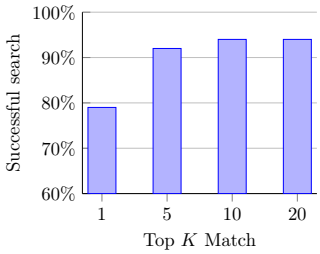


Fig. 3. Top K match for retrieval experiment

consider the percentage of successful cases. Fig. 3 shows the retrieval results for $K \in \{1, 5, 10, 20\}$. Considering the top match ($K = 1$), the search engine successfully retrieved the documents for 79% of the queries. There has been a substantial increase in the top five match to 92%. This increase can be explained by the fact that queries containing low number of notes can retrieve similar pieces and hence the ground truth document will be some where on the top positions of the retrieved list. Only when we become more specific about a theme, it will appear in the top match. Considering $K = 10$, one obtains 94% successful match which indicates that the information retrieval quality is quite good. There is no further increase in the quality of the retrieved list as $K = 20$. If the OMR errors are comparatively smaller, then the document will already be in the top matches of the retrieved list. An inspection on some of the failed queries showed that the original document contains fatal OMR errors. In such cases, obviously the retrieval will not be successful.

V. Conclusion

In this paper we have presented a new melody search engine that searches the noisy OMR data. We evaluated the retrieval quality of the search engine using external sources. The result shows that the retrieval quality is good even though the score inputs were corrupted by the OMR process.

References

[1] A. Rebelo, I. Fujinaga, F. Paszkiewicz, A. R. Marcal, C. Guedes, and J. S. Cardoso, "Optical music recognition: state-of-the-art and open issues," *International Journal of Multimedia Information Retrieval*, vol. 1, no. 3, pp. 173–190, 2012.

[2] D. Bainbridge and T. Bell, "The challenge of optical music recognition," *Computers and the Humanities*, vol. 35, no. 2, pp. 95–121, 2001.

[3] S. Doraisamy and R. Stefan, "A Polyphonic Music Retrieval System Using N-Grams," in *Proceedings of the 5th International Conference on Music Information Retrieval (ISMIR-2004)*, 2004.

[4] S. Balke, S. P. Achankunju, and M. Müller, "Matching Musical Themes based on noisy OCR and OMR input," in *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, 2015, pp. 703–707.

[1] www.multimedialibrary.com/barlow/all_barlow.asp (accessed 08.07.2014)

How can Machine Learning make Optical Music Recognition more relevant for practicing musicians?

Heinz Roggenkemper
Los Gatos, Ca, USA
heinz@roggenkemper.net

Ryan Roggenkemper
Berkeley, CA, USA
rroggenkemper@berkeley.edu

***Abstract*— We describe our experience with building a simple optical music recognition system using machine learning, continue with what we believe the user wants, followed by how machine learning can contribute with better models and through community involvement, and final thoughts.**

Keywords—optical music recognition, machine learning, user expectations, community involvement

I. INTRODUCTION

In recent years optical character recognition (OCR) has moved to be deeply embedded in products and services: scanners often perform OCR directly after scanning so that the PDFs become searchable. Dropbox has automatic OCR as a feature to their business users, and Google offers it in Google Drive (when you open a PDF with Google Docs, OCR is performed in the background). Machine learning and especially the advances of deep learning in areas like image recognition have made this possible. It seems fair to say that OCR has become relevant for many users (sometimes without them being aware that they are using it).

On the first glance OCR and optical music recognition (OMR) seem similar. It would appear that for instance OpenScore (https://openscore.cc/) would be a natural candidate for the application of OMR. However, it focuses on crowdsourced human effort to reach its goal.

Obviously the number of potential users, and the resulting interest and investment differ for OCR and OMR, and there is crucial difference between the two from a user's perspective as well: a page of text that an OCR system processes with 99% accuracy is likely very useful – important services like search documents work, and a user reads the document, the human brain will recognize the meaning of the words and ignore the errors. However, if a violinist is given a one-page score with a 99% pitch accuracy, it is quite possibly useless for her/him – the human ear will neither ignore nor forgive the errors.

II. PROOF OF CONCEPT

As a family of musicians with different skill levels we wanted better access to symbolic music to get scores that fitted our needs. Since not much is available, we tried to use embedded OMR systems with poor results, and found the effort necessary to produce usable symbolic scores through manual work much too high. After learning about how Dropbox had combined computer vision and machine learning to approach OCR [4], we got excited enough to start working on a four-months proof of concept in August 2017, with the following goals:

> Build an OMR system that combines computer vision and machine learning, and achieves an accuracy that is higher than any of the commercial OMR systems that was analyzed in [5] for string quartets. Accuracy in the proof of concept is defined as getting positional pitch and duration right (slurs, accents, dynamics etc. are ignored.) This requires an accuracy of over 90% for pitch and duration. Stretch goal is to achieve an accuracy that is higher than the combined output of multiple sources (95%).

In [5] the comparison was based on Mozart string quartets. We found for string quartet K458 both an IMSLP file and an unrelated MusicXML encoding(IMSLP482550, MusicXML: https://www.gutenberg.org/ebooks/4951). We wrote a set of tools to create individual images from the IMSLP PDF, extracted the labels from the MusicXML file, matched the labels to the images, and checked them carefully. This turned out to be necessary: images were too small, included more than one symbol, or labels were matched to the wrong image. In some cases we found error rates of up to 8%.

In addition, we created synthetic images, mainly for completeness - we wanted all combinations of pitch (G3 to D6) and note length (whole to 1/128) represented. We rendered them both through MuseScore and Finale for difference in appearance.

The individual image files had different sizes, which were normalized to 48x144 pixel, and converted to 8-bit grayscale. The target vector that we extracted from MusicXML contained symbol type (note, rest), pitch, and duration. (The training data for accidentals was derived from synthetic music only.)

In total we used about 10,000 labeled image files of musical symbols (notes, rests, accidentals) in the proof of concept, with about 90% synthetic files. The results from this were:

(1) We were able to create models for classifying the types of symbols, and recognizing both pitch and duration of the notes, and duration of rests, and reached the stretch goal of the proof of concept. SVM models worked well for classification, and for pitch and duration we used 3-layer CNNs, with scikit-learn and TensorFlow as frameworks.

(2) Working with scanned images was a lot harder than working with synthetic notes. That of course is not surprising, but the magnitude of the problems that we encountered was unexpectedly high. When we found it difficult to match the images from IMSLP to the targets we had extracted from MusicXML in the first movement of the Mozart string quartet, we rendered the MusicXML and compared it measure by measure to the IMSLP score. We found 119 differences (none for pitch, all for duration – we ignored slurs/ties and other differences). Based on 4074 notes and 1025 rests in the first movement, the 2.3% difference may not look high, and the differences had little or no musical meaning: almost nobody would notice whether viola and cello play a dotted quarter note, or a quarter note

followed by a 1/8 rest (measure 4), or whether there is one ¼ rest or two 1/8 rests (measure 137).. However, this complicates preparing training data substantially, since automatic matching between image and label is either not possible or can even be wrong.

(3) When we applied the trained model to additional images from a different score (IMSLP 10870) with the same dpi, the accuracy dropped to 82% (the pages did look visibly grainier). It seemed obvious to us that we would need to increase the amount of training material from scanned images very substantially to achieve better results.

(4) Looking back, there is one decision that we now think we got wrong: to focus on positional pitch and to treat accidentals as its own symbol type. It would have been better to treat the accidental as part of the note.

III. What do Musicians want?

We took a step back and, based on our own experience and talks others (members of Ryan's youth orchestra, the conductor of the youth orchestra, amateur musicians and music teachers in the US and Germany, and a composer) to see how musicians currently interact with scores:

- Musicians still buy scores, but everyone uses IMSLP, increasingly on iPads with products like forScore.
- A (small) fraction interacts with symbol music through notation software.
- A fraction of those use OMR software. (The people we talked to use what is bundled with the notation software, and most were with it.)

We came to believe that musicians want services enabled by symbolic music. Imagine the following:

- A musician searches for a score in IMSLP. If she/he wants additional services for the score (like transposing it, play the whole score or sections of the score at a desired speed, allow basic editing), there should be an option to access the result of a high-quality OMR process (like opening a PDF with Google Docs), if the musician is satisfied with the predicted recognition accuracy of the score with an emphasizes pitch accuracy.
- The tool highlights obvious problems (e.g. the note and rest values not adding up to the time signature).

We think that a lot of musicians would use this. Is this a pipe dream? From a technical perspective: no.

IV. What can machine Learning contribute?

Machine learning can support this in the following ways:

(1) Delivering high-accuracy models: [2], [7], and especially [3] have shown that with very sophisticated machine learning models high accuracy can be achieved, competitive with a leading commercial OMR tool.

(2) As soon good models and data exist, additional models (e.g. predicting pitch accuracy for a new page) become easier.

(3) Machine learning systems can improve over time once they effectively and efficiently collect feedback, and learn from it. (An obvious example is Google Maps for driving instructions or identifying areas of interest.)

This would require:

(a) Easily accessible data and pipelines: both [3] and [7] point the way by making their data accessible.

(b) If image augmentation as described in [7] is not sufficient to handle lower quality inputs like scanned images from IMSLP PDFs, then a good way to get to enough training data is needed. In our view, this will require better tools and the involvement of the musician community.

(c) As for (3) we believe that involving the musician community will be key here as well, since feedback needs to be evaluated, for instance to decide whether and how it should be added to the training set, and how retraining is triggered and measured.

Finally, OMR topics seem to not be well-known in the machine learning community. We wonder whether exposing OMR problems in a Kaggle competition could help to change this. An example could be the prediction of page-level accuracy. (Training input would be the page image, the accuracy of the best available model, and set of page level attributes, with required deliverable being a model that not only predicts accuracy, but explains the results as well.)

V. Final remarks

We believe that progress with OMR will require the involvement of the musician and machine learning community. (In that sense, the approach of OpenScore is correct, but in our opinion too limited.) As for what we outlined in III.: it is desirable and feasible [1]. Viability is another matter - in OCR there were Google, Dropbox and others, and we currently do not see their equivalent in this space.

References

[1] Tim Brown: "Change by Design: How Design Thinking Transforms Organizations and Inspires Innovation", 2009.

[2] Jorge Calvo-Zaragoza, Jose J. Valero-Mas, Antonio Pertusa, "End-to-end Optical Music Recognition using Neural Networks", 18th International Society for Music Information Retrieval Conference, Suzhou, China, 2017.

[3] Jorge Calvo-Zaragoza, David Rizo, "End-to-end Neural Optical Music Recognition of Monophonic Scores", Applied Sciences, 2018, 8, 606K.

[4] Brad Neuberg, "Creating a Modern OCR Pipeline Using Computer Vision and Deep Learning", Dropbox Tech Blog, https://blogs.dropbox.com/tech/2017/04/creating-a-modern-ocr-pipeline-using-computer-vision-and-deep-learning/, April 12, 2017

[5] Victor Padilla, Alex McLean, Alan Marsden & Kia Ng. "Improving Optical Music Recognition by Combining Outputs from Multiple Sources". 16th International Society for Music Information Retrieval Conference, 2015

[6] Ana Rebelo, Ichiro Fujinaga, Filipe Paszkeiewicz, Andre R.S. Marcal, Carlos Guedes, Jaime S. Cardoso. "Optical Music Recognition - State-of-the-Art and Open Issues", International Journal of Multimedia Information Retrieval, Vol. 1, No. 3, pp. 173-190, 2012.

[7] Eelen van der Wel, Karen Ulrich, "Optical Music-Recognition with Convolutional Sequence-to-Sequence Models", 18th International Society for Music Information Retrieval Conference, Suzhou, China, 2017

Developing an environment for teaching computers to read music

Gabriel Vigliensoni, Jorge Calvo-Zaragoza, and Ichiro Fujinaga
Schulich School of Music, McGill University, CIRMMT
Montréal, QC, Canada
{gabriel.vigliensonimartin, jorge.calvozaragoza, ichiro.fujinaga}@mcgill.ca

***Abstract*—We believe that in many machine learning systems it would be effective to create a pedagogical environment where both the machines and the humans can incrementally learn to solve problems through interaction and adaptation.**

We are designing an optical music recognition (OMR) workflow system where human operators can intervene to correct and teach the system at certain stages so that they can learn from the errors and the overall performance can be improved progressively as more music scores are processed.

In order to instantiate this pedagogical process, we have developed a series of browser-based interfaces for the different stages of our OMR workflow: image preprocessing, music symbol recognition, musical notation recognition, and final representation construction. In most of these stages we integrate human input with the aim of teaching the computers to improve the performance.

***Index Terms*—optical music recognition, interactive machine learning**

I. A Pedagogy for "Learning Machines"

In this paper, we propose the idea of a *pedagogy for learning machines* as the study of the methods and activities of teaching machines, and the creation of an environment where humans can learn the art of how to teach machines running learning algorithms. In order to achieve this, we first need to understand how humans interact with a machine-learning component and then to build a clever workflow to take advantage of the intelligence of the human and the ability to perform fast calculations of the computer.

Bieger et al. proposed a conceptual framework for teaching intelligent systems [1]. In this framework, the interaction between *teachers* (e.g., a human actor) and *learners* (e.g., a computer system) has the goal of teaching the learning system to gain knowledge about something or about a specific task. As a pedagogical strategy, we hypothesize that by knowing the learner, and how the learner reacts to correction and new input, teachers can adapt their teaching strategies to improve the pedagogy.

II. Teaching Machines How to Read Music Scores

Our aim is to read and extract the content from digitized images of music documents. This process is called optical music recognition (OMR) and, despite more than 50 years of research, it remains a difficult problem.

In order to work at a large scale, we are taking a machine learning-based approach to perform OMR of Medieval and Renaissance music. Instead of using heuristics and features that take advantage of specific characteristics of the documents, we teach the computer to classify the different elements in a music score by training it with a large number of examples for each category to be classified. The computer learns the regularities in these examples and creates a model of the data. Once a model is created, it is used to classify new examples that the computer has not yet seen. In other words, the computer *learns by examples* from the teacher.

The OMR workflow is typically divided into four stages: *image preprocessing*, *music symbol recognition*, *musical notation recognition*, and *final representation reconstruction* [2]. Digitized music scores are the input to the system and image preprocessing is applied to segment the constituent parts of the music document into layers such as music symbols, staff, text, and background. The recognition of the type of music symbols and the analysis of their relationship is achieved once they are isolated and classified in the found layers. Finally, the retrieved musical information is encoded into a machine-readable format.

We want to automate the process of extracting and digitizing the content of music scores. Since we know that this process is not error free, and errors generated in previous steps are carried forward to the next ones, we want to learn about the type of errors that the computer makes in each stage in order to: (i) provide better ground-truth data to improve the performance of the computer and (ii) let users (teachers) of the system understand and know where computers make mistakes in order to modify their behavior.

A. Teaching machines for image segmentation

The first stage in our OMR workflow is *image preprocessing*. In this step, all pixels of the music score image are classified into different, pre-defined layers. Since we need training data as example for recognizing the different layers within an image, and creating ground truth from scratch is onerous and expensive, we have tested a few approaches for teaching the computer to perform image preprocessing. So far, we have found that we can drastically reduce the time and effort needed to build ground truth by preprocessing a small number of images with a pre-existing model, usually a

model learned in pages of similar characteristics. If no model achieves a meaningful result, we use a heuristic method. Then, we correct the coarse errors in the output of the previous stage with a pixel-level editor. In this step, we only amend the major errors in order to have a reasonable set of corrected data. To achieve this, we developed *Pixel.js*, a web-based application designed for correcting the output of pixel-level classification algorithms [3]. We use this tool interactively with a convolutional neural network-based classifier [4] to create ground-truth data incrementally. Finally, we iterate over the two previous steps until the desired performance is achieved. We assume that perfect performance can not be achieved because, at pixel-level, even for humans it is hard to discriminate to what layer a pixel belongs to, especially at the boundaries.

A conventional machine learning approach would work under the assumption that training and tuning will be performed a few times and need not be interactive. Hence, one reasonable strategy for improving supervised learning systems is enabling the user to evaluate a model, then edit its training dataset based on his or her judgments of how the model should improve. Preliminary implementations of these pedagogical strategies and actions have permitted us to reduce the amount of effort when creating ground truth for image preprocessing for OMR.

B. Teaching machines to recognize musical symbols

Our application for the second stage of the OMR workflow, music symbol recognition, is called *Interactive Classifier* (IC). IC is a web-based version of the Gamera classifier [5]. In this stage, the connected components of a specific layer of the original image are automatically grouped into *glyphs*. Then, a human teacher has to manually label the classes of a number of musical glyphs. IC will extract a set of features for describing each of the glyphs, and will classify the data based on the k-nearest neighbors classifier.

IC can be used in an incremental learning fashion [6]. That is, as new data is entered by a human teacher into the system, IC will learn from new information and will accommodate the classes while preserving previously acquired knowledge without building a new classifier. In other words, users of the system can use a previously trained classifier of glyphs and labels for the initial classification. Then, they can manually correct the glyphs that were misclassified and perform a reclassification. By repeating this process, IC will learn the corrections at each iteration and will build a better classifier until the teacher is satisfied with the results.

An interesting characteristic of IC is that how well the machine learns depends on how well the human teaches it. In fact, the human, through interaction, can gradually learn how to teach the machine better.

C. Non-pedagogical OMR stages

The last two stages of our OMR workflow, *musical notation recognition* and *final representation construction* have a common interactive breakpoint for visualizing and correcting the output of the automatized OMR process. This human-driven checkpoint is embedded as a web-based interface called *Neume Editor Online* (Neon) [7]. Neon allows a user to inspect differences between the original music score image and the rendered version of the output of the OMR process. By visual inspection of the two overlaid scores, the user can observe their difference and manually add, edit, or delete music symbols in the browser. So far, however, corrections entered by the user are not fed back into the learning system, but they change the encoded music file output.

D. Our OMR workflow management system

Since our workflow requires a human operator to teach the learning system, we need to be able to create interactive checkpoints where the system stops a process and waits for user input. As a result, all the constituent parts of our OMR workflow are handled by Rodan, a distributed workflow management system [8] that allows to specify *interactive* and *non-interactive* tasks. In case more efficient implementations of OMR tasks become available, Rodan also allows wrapping and incorporating them into a compatible workflow.

III. Final remarks

The end goal of our project is to create a final music representation that is browsable and searchable by humans and computers by many different means. We envision this interface as an intelligent, music-score-searching tool for the 21st century. We hope that new tools and infrastructure, in combination with the proper teaching strategies and tactics developed by human teachers in the interfaces for training the OMR system, will enable the end-to-end recognition and encoding of music from music score images.

References

[1] J. Bieger, K. R. Thórisson, and B. R. Steunebrink, "The pedagogical pentagon: A conceptual framework for artificial pedagogy," in *International Conference on Artificial General Intelligence*, ser. Lecture Notes in Computer Science, vol. 10414, T. Everitt, B. Goertzel, and A. Potapov, Eds. Springer, 2017, pp. 212–222.

[2] A. Rebelo, I. Fujinaga, F. Paszkiewicz, A. R. S. Marcal, C. Guedes, and J. S. Cardoso, "Optical music recognition: State-of-the-art and open issues," *International Journal of Multimedia Information Retrieval*, vol. 1, no. 3, pp. 173–190, Oct 2012.

[3] Z. Saleh, K. Zhang, J. Calvo-Zaragoza, G. Vigliensoni, and I. Fujinaga, "Pixel.js: Web-based pixel classification correction platform from ground truth creation," in *Proceedings of the 12th IAPR International Workshop on Graphics Recognition*, Kyoto, Japan, 2017.

[4] J. Calvo-Zaragoza, F. J. Castellanos, G. Vigliensoni, and I. Fujinaga, "Deep neural networks for document processing of music score images," *Applied Sciences*, vol. 8, no. 5, pp. 654–674, 2018.

[5] M. Droettboom, K. MacMillan, and I. Fujinaga, "The Gamera framework for building custom recognition systems," in *Proceedings of the 2003 Symposium on Document Image Understanding Technologies*, Greenbelt, MD, 2003, pp. 275–286.

[6] R. Polikar, L. Upda, S. S. Upda, and V. Honavar, "Learn++: An incremental learning algorithm for supervised neural networks," *IEEE Transactions on Systems, Man, and Cybernetics—Part C: Applications and Reviews*, vol. 31, no. 4, pp. 497–508, 2001.

[7] G. Burlet, A. Porter, A. Hankinson, and I. Fujinaga, "Neon.js: Neume editor online," in *Proceedings of the 13th International Society for Music Information Retrieval Conference*, 2012, pp. 121–126.

[8] A. Hankinson, "Optical music recognition infrastructure for large-scale music document analysis," Ph.D. dissertation, McGill University, Montréal, QC, 2015.