PROTOTYPING FULL-PIPELINE OPTICAL MUSIC RECOGNITION WITH MUSCIMARKER

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

Matthias Dorfer
Johannes Kepler Universität
Institute of Computational Perception
matthias.dorfer@jku.at

ABSTRACT

We present MUSCIMarker, an open-source workbench for developing Optical Music Recognition (OMR) systems from image preprocessing to MIDI export. It is built around the notation graph data model of the MUSCIMA++ dataset for full-pipeline OMR. The system is transparent and interactive, enabling the user to visualize, validate and edit results of individual OMR stages. It is platform-independent, written purely in Python, and can work offline. We demonstrate its value with a prototype OMR system for musical manuscripts that implements the recognition pipeline, up to playing the recognition outputs through MIDI. The audience will interact with the program and can test an integrated OMR system prototype.

1. INTRODUCING MUSCIMARKER

The task of recovering symbolic musical information such as MIDI from the image of the written score has been addressed by the field of Optical Music Recognition (OMR) for half a century. Much more music has been written than has been recorded, which makes OMR important to making a large portion of musical heritage accessible to both the professional and the general public. However, OMR has not yet been able to deliver satisfactory solutions for anything but high-quality scans of printed music – much less for manuscripts, even though more pieces probably remain in handwritten form than have been typeset, both from long-dead and contemporary composers. We can therefore expect the development of OMR systems to continue for some time. OMR systems also need diangnosing with respect to the final output, as well as in the individual stages of the typical recognition pipeline (see [1]). At the same time, although one dataset is now available that provides ground truth sufficient for experiments on the full OMR pipeline (MUSCIMA++ [5]), the variability of sheet music means further data acquisition is still necessary.

To address these needs, we present MUSCIMarker: a workbench for OMR groundtruthing and prototyping.

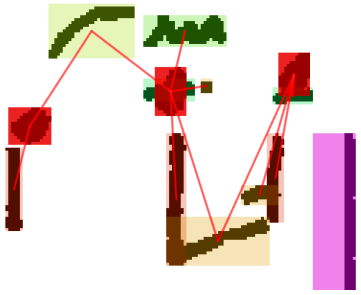

Figure 1. Notation graph: notation symbols are vertices; edges encode which symbols have to be interpreted in relationship to each other. Staff symbols have been removed for clarity.

1.1 Data model

OMR aims to recover the "musical content", which usually means the piano roll representation. In terms of notes, this can be thought of in terms of recovering *notes in time*: their pitches, durations, and onsets. These properties can be inferred deterministically from the the *notation graph* of the musical score [5]. The vertices of this graph are the individual symbols (with properties such as symbol class and position on the page), the edges are the logical relationships between symbols. Each encoded note is represented by a notehead-class vertex, and its pitch, duration and onset can be decoded from the relationships of the notehead to other symbols that affect these parameters, such as stems, stafflines, or accidentals. A simplified example of a notation graph is in Figure 1.

Because the data model for music scores is an open problem, MUSCIMarker also allows customizing it.

1.2 MUSCIMarker Features

The core functionality of MUSCIMarker is manipulating the notation graph for a given musical score. Advances in OMR can then be integrated into MUSCIMarker as "automated helpers".

Its closest "cousin" is probably the Aletheia system for OCR groundtruthing [4]. However, music notation needs a richer data model than the document layout and glyph model that Aletheia provides. The related gamera [1] OMR

[1] http://gamera.informatik.hsnr.de

toolkit does not at all provide the interactive functionality necessary for groundtruthing and correcting errors of automated subsystems integrated into the tool.

MUSCIMarker implements the following feature sets:

- Efficient tools for creating and editing the notation graph, incl. interactive binarization,
- User activity tracking, both for optimizing "bottlenecks" in usability and for assessing experimental subsystems in terms of error correction times,
- Automated notation graph validation.

To obtain the piano roll representation, MUSCIMarker then implements pitch/duration/onset inference for the particular definition of the notation graph used in MUSCIMA++, and can play back MIDI for the user.

On top of this, we have trained prototype OMR subsystems that perform symbol detection and notation graph reconstruction from symbols, so that in principle a full OMR pipeline is available (even though these prototypes' performance leaves something to be desired).

All functionality is available for the audience to try.

2. PROTOTYPE OMR SYSTEM OVERVIEW

Recent progress in image processing methods using convolutional networks and the release of the new MUSCIMA++ dataset [5] have together made building a viable OMR solution for handwritten musical scores a more realistic proposition, although progress has so far been mainly focused on staffline removal [2, 3]. We include a prototype of an OMR solution integrated into MUSCIMarker.

We build the notation graph in two steps: finding the vertices of the notation graph, which means detecting notation symbols, and recovering their relationships, the edges of the graph.

Symbol detection has models trained for symbols that participate in pitches, durations, and onsets. For each symbol, the detector is a U-net (see Figure 2) [6]; each class is trained independently. Staffline detection is handled in exactly the same manner as other symbols, and staffline removal is not necessary for detecting other classes of symbols. The FCN outputs a probability mask; we threshold this mask at 0.3 and then detect symbols using connected components. The detector needs a GPU to achieve reasonable speed, and thus runs remotely.

Relationships between symbols are then inferred. There are two types of relationships: *attachment* and *precedence*. Attachments are syntactic relationships between primitives, such as that between a notehead and a stem or staffline, which are necessary for obtaining pitch and duration; precedence edges are used to compute onsets. A manual parser is provided that simply adds all edges among a manually selected subset of vertices that are possible based on the symbols' classes. For attachments, a probabilistic parser is also trained, so that one can work with larger selections; the probabilistic parser is a binary classifier that classifies each (ordered) pair of symbols according to the difference between their bounding boxes and the symbol classes.

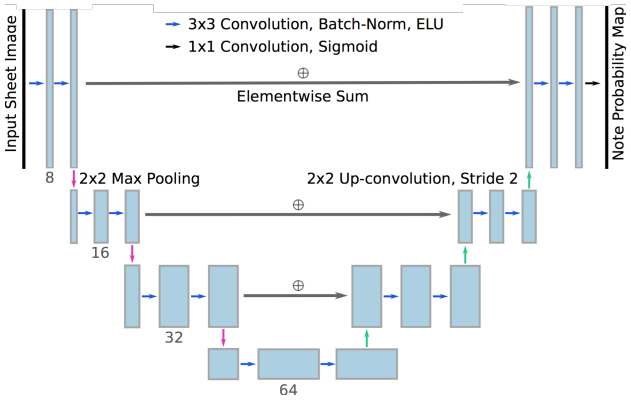

Figure 2. Detector network architecture.

3. CLOSING REMARKS

We hope that the presented MUSCIMarker tool for OMR prototyping and groundtruthing will be valuable to OMR researchers, and by implication to future OMR users.[2] We look forward to audience feedback, to guide further development so that we come closer to bringing musical manuscripts into the digital fold.

4. REFERENCES

[1] 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*, 1(3):173–190, Mar 2012.

[2] Antonio-Javier Gallego and Jorge Calvo Zaragoza. Staff-line removal with selectional auto-encoders. *Expert Systems with Applications*, 89:138 – 148, 2017.

[3] Jorge Calvo Zaragoza, Antonio Pertusa, and Jose Oncina. Staff-line detection and removal using a convolutional neural network. *Machine Vision and Applications*, pages 1–10, 2017.

[4] Christian Clausner, Stefan Pletschacher, and Apostolos Antonacopoulos. Aletheia - An Advanced Document Layout and Text Ground-Truthing System for Production Environments. In *2011 International Conference on Document Analysis and Recognition, ICDAR 2011, Beijing, China, September 18-21, 2011*, pages 48–52. IEEE Computer Society, 2011.

[5] Jan Hajič, jr. and Pavel Pecina. In Search of a Dataset for Handwritten Optical Music Recognition: Introducing MUSCIMA++. *ArXiv e-prints*, March 2017.

[6] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-Net: Convolutional Networks for Biomedical Image Segmentation. In *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015: 18th International Conference, Munich, Germany, October 5-9, 2015, Proceedings, Part III*, pages 234–241, Cham, 2015. Springer International Publishing.

[2] https://github.com/hajicj/MUSCIMarker