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 data set *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* data set of synthetic music scores together with ground truth [2] 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* DATA SET

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 [2] annotations. Every labeled object now has an *onset* tag that tells the start beat of the the given object. All note heads 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* data set is that it is synthetic. We are currently working on a much smaller data set, 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 data set, we observed that the detector was unable to find anything on the testing set despite that the loss function converged. This lead 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 data set).

III. FURTHER RESEARCH ON DEEP WATERSHED DETECTION

A. Augmenting inputs

DeepScores, unlike many academic data sets, is extremely unbalanced. In fact, the most common class (note head

*Equal contribution

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

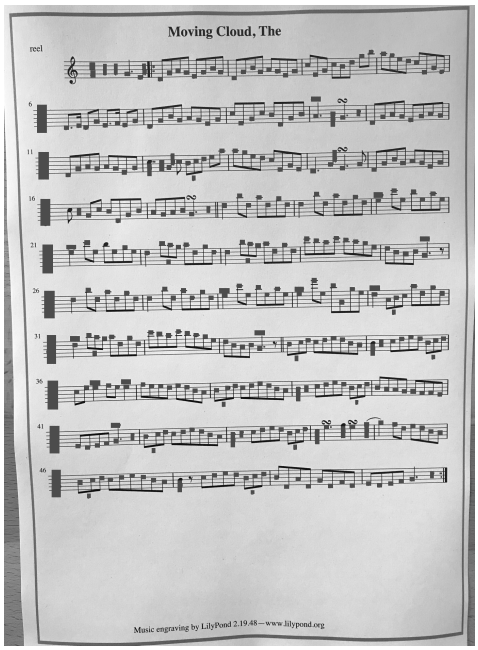

Fig. 2. Preliminary results of our model (grey boxes) on a photo of a printed sheet. While not perfect (for example, our model misses the clef in the first row), they already look promising.

black) 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 data synthesis, where we locate rare symbols in the data set, 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 data set. 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* data set. The two approaches are complementary and preliminary results show improvement, though more investigation is needed: overfitting on extreme 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.

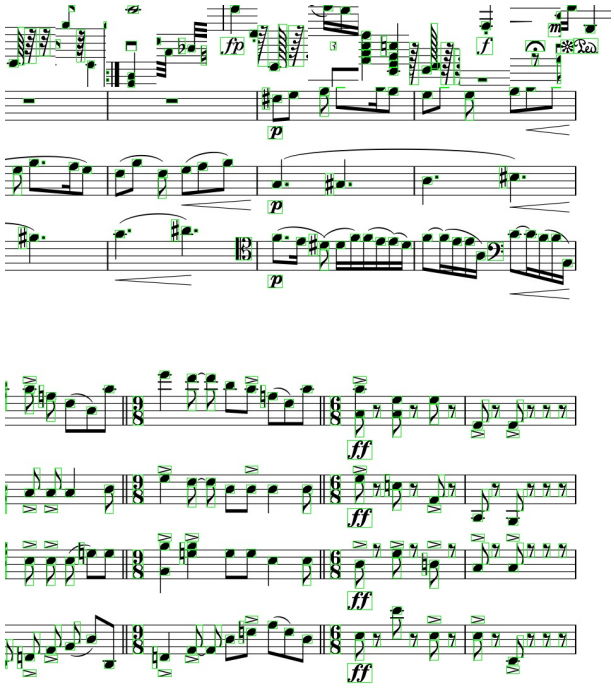

Fig. 3. A musical score where 12 small images have been augmented at the top of 7 regular staves. The bounding boxes are marked in green.

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. Lrwald, 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).