END-TO-END OPTICAL MUSIC RECOGNITION USING NEURAL NETWORKS

Jorge Calvo-Zaragoza
Centre for Interdisciplinary Research in Music
Media and Technology, McGill University
Montreal, QC, Canada

Jose J. Valero-Mas, Antonio Pertusa
Software and Computing Systems
University of Alicante
Alicante, Spain

ABSTRACT

This work addresses the Optical Music Recognition (OMR) task in an end-to-end fashion using neural networks. The proposed architecture is based on a Recurrent Convolutional Neural Network topology that takes as input an image of a monophonic score and retrieves a sequence of music symbols as output. In the first stage, a series of convolutional filters are trained to extract meaningful features of the input image, and then a recurrent block models the sequential nature of music. The system is trained using a Connectionist Temporal Classification loss function, which avoids the need for a frame-by-frame alignment between the image and the ground-truth music symbols. Experimentation has been carried on a set of 90,000 synthetic monophonic music scores with more than 50 different possible labels. Results obtained depict classification error rates around 2 % at symbol level, thus proving the potential of the proposed end-to-end architecture for OMR. The source code, dataset, and trained models are publicly released for reproducible research and future comparison purposes.

1. INTRODUCTION

Large-scale analysis of music is of great interest, and so many computational tools have been developed for such purpose. Quite often, the bottleneck for exploiting these ideas is the lack of large corpora of symbolic music.

The transcription of sheet music into some machine-readable format can be carried out manually. However, the complexity of music notation inevitably leads to burdensome software for music score editing, which makes the whole process very time-consuming and prone to errors. As a consequence, the development of automatic transcription systems for musical documents is gaining importance over the last years.

The field devoted to address this task is known as Optical Music Recognition (OMR) [1]. Typically, an OMR tool takes an image of a music score and provides its symbolic content encoded in some structured digital format such as MEI or MusicXML. Unfortunately, OMR is a challenging problem, and results have not been very promising so far [18].

The process of automatically recognizing the content of a music score is complex, and therefore the workflow of an OMR system is very extensive. Previous proposals related to this task focus on specific aspects of the pipeline, such as the binarization of the image [14], the detection of the staves [2], the separation between lyrics and music [3], the staff-line removal [8]—which may be even considered as a task by itself [7]—or the classification of isolated symbols [17]. It therefore comes as no surprise that no work have directly addressed the whole OMR process for modern western notation. We only find full recognition proposals for old music [5,15,16] that, in spite of involving music notation, entails a very different challenge.

One of the practical aspects that constrains end-to-end OMR research is the difficulty of obtaining an aligned dataset containing the labeled music symbols along with their exact position in the image of the score. Note that, from a musical perspective, it is not necessary to retrieve the exact position of each music symbol in the image since the important information is the succession of the music figures. Thus, it seems interesting to tackle the OMR task in an holistic fashion, in which the output is directly the sequence of symbols present in the score image disregarding their exact position in pixels.

Our work aims at setting the basis towards the development of systems that can directly work with a greater part of the OMR workflow. For that, we propose the use of recurrent neural networks, which have been applied with great success to many sequential recognition task such as speech recognition [11], handwriting recognition [12], or automatic music transcription [20]. The premise is that the network works on a single staff section, much in the same way as most Optical Character Recognition systems focuses on recognizing words appearing in a given line image [21,23].

The traditional limitation of such type of networks is that they require a strongly-aligned training set, i.e., the network has to be provided with the desired output of the recurring block for every single input frame of the image. This constraint has typically led to consider other sequential models such as hidden Markov Models, which can be trained with just pairs of input images and tran-

© Jorge Calvo-Zaragoza, Jose J. Valero-Mas, Antonio Pertusa. Licensed under a Creative Commons Attribution 4.0 International License (CC BY 4.0). **Attribution:** 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.

script sequences. Nonetheless, Graves et al. [10] proposed a method to train recurrent networks with unaligned data known as Connectionist Temporal Classification (CTC). The CTC is actually a loss function that focuses on the desired output sequence, regardless of which frames output each symbol.

For the precise case of this work, we rely on the Convolutional Recurrent Neural Network (CRNN) architecture for scene text recognition proposed by Shi et al. [19]. A CRNN is a deep neural network that comprises a series of convolutional layers, which focus on learning a suitable representation of the input image, followed by recurrent layers, which deal with the sequential nature of the task. In order to jointly train the network in an end-to-end fashion, the CTC loss function is considered.

Besides text recognition, Shi et al. also evaluated CRNN with a small number of music scores, just to assess its capabilities for any sequence-based task. Taking this work as a starting point, we further study the potential of the mentioned end-to-end CRNN model for the case of OMR. More precisely, our contributions are: (i) the redesign and optimization of the original CRNN architecture for this particular task; (ii) a thorough and quantitative assessment of the proposed architecture in terms of a large collection of more than 90,000 monophonic music scores.

The rest of the paper is structured as follows: Section 2 describes the details of the corpus created for this work; Section 3 describes the end-to-end model proposed; the evaluation procedure as well as the results obtained are shown and discussed in Section 4; finally, Section 5 concludes the work and proposes future lines to address.

2. CORPUS GENERATION

For assessing the proposed scheme we generated a set of monophonic score images together with their ground-truth annotations disregarding any frame-level alignment for the case of end-to-end training. This set contains 94,984 random sequences from a vocabulary of 52 Common Western Music Notation symbols: music notes from C4 to E5 (10 pitches), four possible note durations (half, quarter, eighth, and sixteenth) and their four respective silences, three time signatures (3/4, 4/4, and 6/8), accidentals (sharp, flat, and natural), the treble clef, and the bar line.

All the scores follow this structure: an initial clef; a set of alterations for the key of the piece; the time signature; the music content, being always the bar line annotated as it constitutes a symbol to be recognized. Note that bar lines are not randomly placed in the score but in their corresponding positions at the end of each complete bar.

The length of the generated sequences is random, with a minimum length of 4 symbols and a maximum of 37. Figure 1 shows a histogram of the length of the produced sequences.

The generation of the music content is random, i.e., no restriction is imposed about the pitch interval between two consecutive notes or their respective duration. Similarly, accidentals are randomly applied to further increase the variability in the scores. Given a sequence of music symbols we generated the image scores with the music engraving software Lilypond [1]. Figure 2 shows two examples of music scores along with their ground-truth annotations.

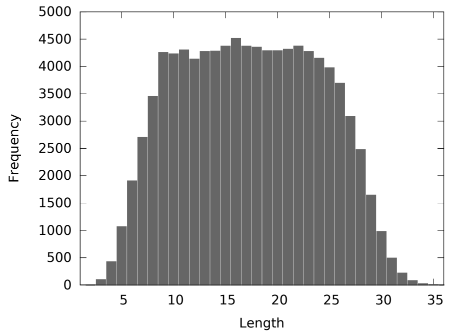
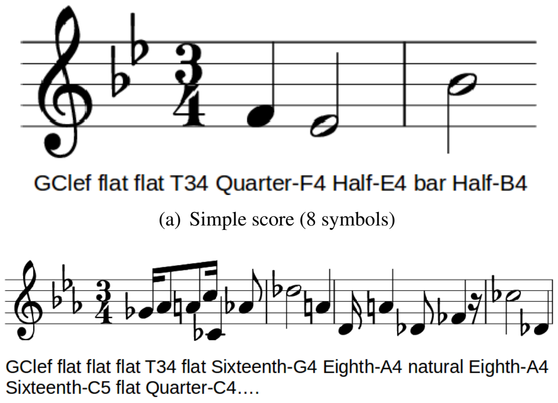


Figure 1. Histogram of the length of the sequences of the corpus.



GClef flat flat T34 Quarter-F4 Half-E4 bar Half-B4
GClef flat flat flat T34 flat Sixteenth-G4 Eighth-A4 natural Eighth-A4 Sixteenth-C5 flat Quarter-C4....

(a) Simple score (8 symbols)

(b) Challenging score (34 symbols)

Figure 2. Example of scores depicting different levels of difficulty from our collection, along with its associated ground-truth.

3. FRAMEWORK

Our OMR approach is based on a Convolutional Recurrent Neural Network (CRNN) which takes as input an image of a monophonic staff section and directly outputs the sequence of music symbols, with no previous symbol segmentation or staff-line removal process. A conceptual scheme is illustrated in Figure 3.

Before the actual CRNN, we assume that a preprocessing step identifies and segments the different monophonic staff sections from the initial image for processing them independently. While this may be seen as a strong assumption, there exist algorithms in the literature that successfully address this task [6]. Once this monophonic staff section is segmented, the resulting image is normalized (pixel values between 0 and 1), rescaled to an aspect ratio of 1:4

[1] http://lilypond.org/

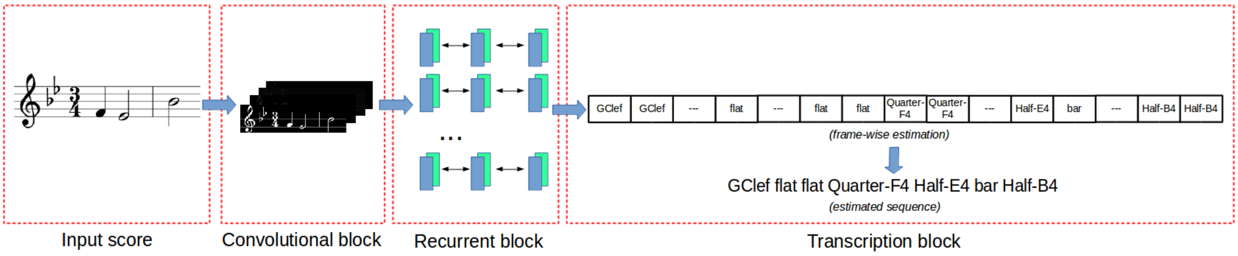


Figure 3. Conceptual scheme of the proposed approach. The input score is processed with a series of convolutional filters; the resulting features are then processed by the recurrent layers to model the temporal context of the piece; a frame-wise transcription using CTC is performed to obtain the estimation in an end-to-end fashion.

(i.e., the width is four times the height), and used as input to the CRNN. We established that this ratio is adequate for the task at issue by means of informal testing.

Table 1 shows the specific details of the proposed CRNN architecture, whose configuration and parameterization were determined experimentally. First, the image is processed with a series of convolutional layers which use Rectifier Linear Unit (ReLU) activation functions, followed by max pooling layers. Then, the output of the convolutional block is reshaped to serve as input to a recurrent neural network block, which is composed of three Bidirectional Long-Short Term Memory (BLSTM) networks [9, 13] with 256 hidden units. Finally, a fully-connected layer with a SoftMax activation function is added to retrieve the most likely class of each frame.

The CRNN model is trained using a batch size of 32 samples (i.e., 32 monophonic staff sections), RMSprop as the gradient descent method, and the aforementioned CTC as the loss function. We set 20 epochs for the training of the model and selecting the configuration that minimizes the validation error.

Note that the output of the CRNN is a framewise prediction that must be processed to obtain the actual output symbol sequence. However, this process is very straightforward because the CTC loss function forces the network to predict a *blank* symbol to indicate the separation between consecutive symbols [10].

4. EVALUATION

4.1 Partitions

We split the generated corpus in three fixed partitions: training and validation, which are meant to train the model and select the most appropriate hyper-parameters of the network, and a test partition to eventually assess the performance of the system. These sets represent the 60 %, 20 %, and 20 % out of the total set of available scores, respectively.

Table 2 describes these partitions in terms of the number of scores, measures, and running symbols in each of them. It must be noted that at least one element of the vocabulary appears in all the partitions, and so there are no out-of-vocabulary elements.

4.2 Metrics

In order to assess the performance of the proposed method we consider three metrics which allow the evaluation at different levels:

- *Score-level error rate* (S_e): ratio of scores that are not correctly recognized in their entirely (i.e., contain at least one error amongst the estimated ones).
- *Edit distance* (E_d): average number of edit operations to convert the predicted sequence into the ground-truth one.
- *Normalized edit distance* (E_{d_N}): same as the *Edit distance* metric but normalizing each sequence by its length.

Note that the relevance of each metric depends on the final scenario. If a totally autonomous system is pursued, it is important to pay attention to the score-level error. However, quite often it is assumed that an expert user will supervise the output of the system because guaranteeing a error-free model is not feasible [4]. In this case, therefore, it is more interesting to measure the errors at the symbol level, which is more related to the number of corrections to be made.

4.3 Results

Input images must be resized to fixed dimensions for the input of the network. As mentioned earlier, an aspect ratio of 1:4 was chosen. Thus, we have experimented with values involving 40×160, 50×200, and 60×240.

For each case, network parameters are optimized by means of the training set, while the validation set is used to find the most appropriate epoch to stop. The metric chosen to determine the performance after each epoch during training is the normalized edit distance (E_{d_N}).

Once a model is trained, predictions are made on the samples of test set. Table 3 shows the results of our series of experiments in terms of the three figures of merit previously described.

An initial remark to begin is that all input sizes behave similarly. In all the cases, a remarkable performance at symbol level is attained, with figures lower than 0.6 and $4\,\%$ for E_d and E_{d_N}, respectively. It is true, however, that

Block	Configuration			

Convolutional	Conv(64,3,3)
	MaxPool (2,2)
		Conv(128,3,3)
		MaxPool (2,2)
			Conv(256,3,3)
			Conv(256,3,3)
			MaxPool(2,1)
				Conv(512,3,3)
				Conv(512,3,3)
				MaxPool(2,1)

Recurrent	BLSTM (256)	BLSTM (256)	BLSTM (256)	FC (52)

Table 1. Description of the CRNN architecture considered. Notation Conv(f,w,h) stands for a layer with f convolution operators of size $w \times h$ pixels followed by a ReLU activation function. MaxPool(w,h) stands for the max-pooling operator of dimensions $w \times h$ pixels, BLSTM(n) represents a Bidirectional Long-Short Term Memory unit with n hidden layers, and FC(n) is a fully-connected layer of n neurons followed by a SoftMax activation function.

	Training	Validation	Test
Scores	56 991	18 996	18 997
Measures	125 971	41 883	41 986
Symbols	989 744	329 802	330 092

Table 2. Statistics of the partitions used in this work, reporting the number of scores, the number of measures, and the number of running symbols.

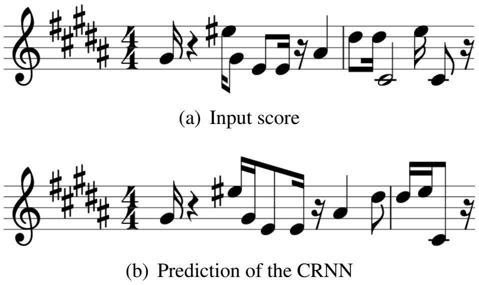
	Metric		
Input image	S_e (%)	E_d	E_{d_N} (%)
40 × 160	27.30	0.52	3.01
50 × 200	29.79	0.54	3.12
60 × 240	22.37	0.37	2.16

Table 3. Performance achieved on the test partition with respect to the shape of the input image.

the score-level error rates are much higher. That is, quite often there is at least one incorrectly recognized symbol in each score sequence.

Best results are obtained using images of 60 × 240. In that case, a symbol-level error rate of 22.37 % is attained, with an average of 0.37 symbol-level errors per score (2.16 % of the symbols if lengths are taken into account). This means that less than one symbol has to be corrected to obtain the actual score, on average. An example of prediction results depicting representative transcription errors is illustrated in Figure 4. Note how some of these error change the arrangement of the beamed groups, as the predicted sequence does not fulfill time signature constraints.

Clearly, these results reflect that the proposed framework allows recognizing accurately the symbols of monophonic scores in an end-to-end manner. In turn, the approach is not so reliable to optimize the number of perfectly recognized images, regardless of the number of errors. However, it has to be considered that some music symbols of the generated scores have vertical overlapping, as can be seen in the first note C from Figure 2. When this happens, the order of the symbolic sequence might not perfectly align with the order of the symbols in the image, thereby introducing noise in the samples.


(a) Input score

(b) Prediction of the CRNN

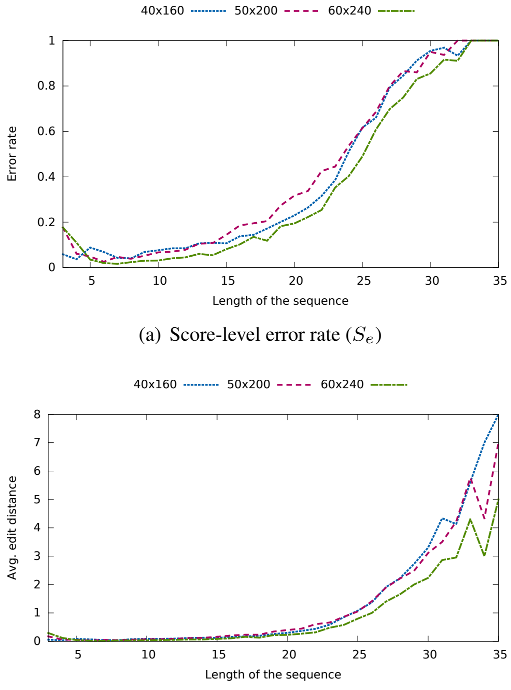
Figure 4. An example of prediction with errors ($E_d = 3$, $E_{d_N} = 11.53$) obtained in our experiments.

As discussed in Sect. 1, there are no previous approaches dealing with the OMR task in an end-to-end way and, therefore, there is no feasible comparison in this work. Nevertheless, it is our hope that these results will establish a new way of approaching OMR.

4.4 Further Analysis

In this section we further analyze some details of the experiments carried out.

First we intend to measure the performance of the models with respect to the size of the input sequence. Clearly, the size of the sequence has a direct impact on the ability of the models to recognize all their symbols. It is expected that the greater the number of symbols in the score, the worse performance the models attain. Figure 5 shows the performance curves as a function of the size of the sequences. On the one hand, Figure 5(a) reports the error rate curve, which depicts that the performance gets dramatically worsen from sequences of 10-15 symbols, depending on the model. On the other hand, Figure 5(b) shows the curve of the edit distance, for which it is observed that the average number of editing operations to correct a sequence predicted by the model is less than 1 up to 25 symbols. The interesting remark about these curves is that they allow us to conclude that in relatively short sequences, the models can obtain an almost optimal performance. Fortunately, the scores can be further subdivided into bars, which have a limited number of symbols. Therefore, it might be interesting to address the problem by first performing a segmentation of measures, for which there already exists successful algorithms [22] as previously mentioned.



(a) Score-level error rate (S_e)

(b) Edit distance (E_d)

Figure 5. Performance attained by the models with respect to the length of the input sequences.

Finally, it is interesting to analyze the convergence for each considered model, since that is an indicator of the representation capacity and the difficulty with which they learn the task. Figure 6 shows the normalized edit distance on the validation set as a function of the number of training epochs. It is observed that all models follow a similar trend, in which there is a drastic decrease in the first 6 epochs. After that, models begin to need more epochs to improve their results, reaching convergence (except for minor fluctuations) around 12 epochs. We can therefore say that all models have a similar representation capabilities, although it has been demonstrated in the previous section that the model that accepts 60 × 240 images has a greater generalization ability. In addition, the low number of required epochs indicate that the models are able to learn the task quickly.

5. CONCLUSIONS

This work addresses the Optical Music Recognition task in an end-to-end fashion with the use of a Convolutional Recurrent Neural Network (CRNN). We have redesigned the architecture from Shi et al. [19] for OMR using a large collection of over 90,000 synthetic scores generated through Lilypond, a music engraving system. As the network is trained using a Connectionist Temporal Classification loss function, the music symbols do not need to be aligned with the pixels of the original images.

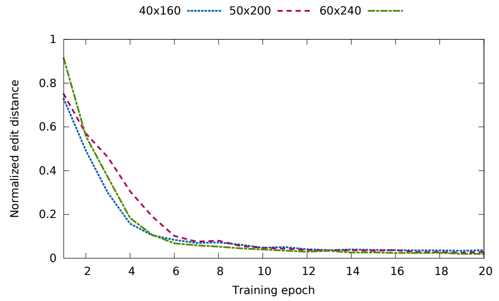


Figure 6. Validation performance (normalized edit distance) with respect to the number of training epochs.

The CRNN topology and hyper-parameters were experimentally adjusted for the task at hand, obtaining remarkably low error rates with the evaluated corpus. The network converges quickly and an average edit distance of 0.37 is obtained using as input 60 × 240 images.

In order to increase the accuracy of the proposed method, those scores containing temporal overlappings could be removed from the corpus. However, the ultimate goal of OMR is to detect music symbols in polyphonic scores. This is a challenging task using CRNN as it implies to extend CTC for multi-label classification, which stands as future work to explore and study.

Another evident future work line is to train the network with real scores. Synthetic data could be used as a basis by adding noise and transformations such as rotation or scaling for a preliminary experimentation as in [19], but ideally a large real corpus should be used instead. Currently there are no large datasets containing labeled images of real scores, but an end-to-end annotation of the data is straightforward as it does not requires the symbols to be aligned with the image pixels.

Finally, note that one of the main advantages of the proposed neural-based approach is that alternative notations could be recognized by just changing the corpus and retraining the model. This opens a path for research in research of ancient music recognition written in, for instance, mensural or neume notation, among others.

6. REPRODUCIBILITY

For reproducibility purposes, the source code, trained models, and considered data have been publicly released at `http://grfia.dlsi.ua.es/gen.php?id=software`.

7. ACKNOWLEDGEMENT

This work was partially supported by the Social Sciences and Humanities Research Council of Canada, and Universidad de Alicante through FPU program (UAFPU2014–5883) and grant GRE-16-04.

8. REFERENCES

[1] D. Bainbridge and T. Bell. The challenge of optical music recognition. *Computers and the Humanities*, 35(2):95–121, 2001.

[2] V. Bosch, J. Calvo-Zaragoza, A. H. Toselli, and E. Vidal. Sheet music statistical layout analysis. *International Conference on Frontiers in Handwriting Recognition 2016*, 2016.

[3] J. A. Burgoyne, Y. Ouyang, T. Himmelman, J. Devaney, L. Pugin, and I. Fujinaga. Lyric extraction and recognition on digital images of early music sources. In *Proceedings of the 10th International Society for Music Information Retrieval Conference*, pages 723–727, 2009.

[4] J. Calvo-Zaragoza and J. Oncina. An efficient approach for interactive sequential pattern recognition. *Pattern Recognition*, 64:295–304, 2017.

[5] J. Calvo-Zaragoza, A. H. Toselli, and E. Vidal. Early handwritten music recognition with hidden markov models. In *15th International Conference on Frontiers in Handwriting Recognition, ICFHR 2016, Shenzhen, China, October 23-26, 2016*, pages 319–324, 2016.

[6] V. B. Campos, J. Calvo-Zaragoza, A. H. Toselli, and E. Vidal-Ruiz. Sheet music statistical layout analysis. In *15th International Conference on Frontiers in Handwriting Recognition, ICFHR 2016, Shenzhen, China, October 23-26, 2016*, pages 313–318, 2016.

[7] C. Dalitz, M. Droettboom, B. Pranzas, and I. Fujinaga. A comparative study of staff removal algorithms. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(5):753–766, 2008.

[8] T. Géraud. A morphological method for music score staff removal. In *Proceedings of the 21st International Conference on Image Processing (ICIP)*, pages 2599–2603, Paris, France, 2014.

[9] F. A. Gers, N. N. Schraudolph, and J. Schmidhuber. Learning precise timing with lstm recurrent networks. *J. Mach. Learn. Res.*, 3:115–143, March 2003.

[10] A. Graves, S. Fernández, F. Gomez, and J. Schmidhuber. Connectionist temporal classification: Labelling unsegmented sequence data with recurrent neural networks. In *Proceedings of the 23rd International Conference on Machine Learning*, ICML '06, pages 369–376, New York, NY, USA, 2006. ACM.

[11] A. Graves, A.-r. Mohamed, and G. Hinton. Speech recognition with deep recurrent neural networks. In *IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 6645–6649. IEEE, 2013.

[12] A. Graves and J. Schmidhuber. Offline handwriting recognition with multidimensional recurrent neural networks. In *Advances in neural information processing systems*, pages 545–552, 2009.

[13] S. Hochreiter and J. Schmidhuber. Long short-term memory. *Neural Comput.*, 9(8):1735–1780, November 1997.

[14] T. Pinto, A. Rebelo, G. A. Giraldi, and J. S. Cardoso. Music score binarization based on domain knowledge. In *Pattern Recognition and Image Analysis - 5th Iberian Conference, IbPRIA 2011, Las Palmas de Gran Canaria, Spain, June 8-10, 2011. Proceedings*, pages 700–708, 2011.

[15] L. Pugin. Optical music recognitoin of early typographic prints using hidden markov models. In *ISMIR 2006, 7th International Conference on Music Information Retrieval, Victoria, Canada, 8-12 October 2006, Proceedings*, pages 53–56, 2006.

[16] C. Ramirez and J. Ohya. Automatic recognition of square notation symbols in western plainchant manuscripts. *Journal of New Music Research*, 43(4):390–399, 2014.

[17] A. Rebelo, G. Capela, and J. S. Cardoso. Optical recognition of music symbols - A comparative study. *IJDAR*, 13(1):19–31, 2010.

[18] 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*, 1(3):173–190, 2012.

[19] B. Shi, X. Bai, and C. Yao. An end-to-end trainable neural network for image-based sequence recognition and its application to scene text recognition. *Computing Research Repository*, abs/1507.05717, 2015.

[20] S. Sigtia, E. Benetos, and S. Dixon. An end-to-end neural network for polyphonic piano music transcription. *IEEE/ACM Transactions on Audio, Speech and Language Processing (TASLP)*, 24(5):927–939, 2016.

[21] A. H. Toselli, V. Romero, M. Pastor, and E. Vidal. Multimodal interactive transcription of text images. *Pattern Recognition*, 43(5):1814–1825, 2010.

[22] G. Vigliensoni, G. Burlet, and I. Fujinaga. Optical measure recognition in common music notation. In *Proceedings of the 14th International Society for Music Information Retrieval Conference, ISMIR 2013, Curitiba, Brazil, November 4-8, 2013*, pages 125–130, 2013.

[23] P. Voigtlaender, P. Doetsch, and H. Ney. Handwriting recognition with large multidimensional long short-term memory recurrent neural networks. In *15th International Conference on Frontiers in Handwriting Recognition, ICFHR 2016, Shenzhen, China, October 23-26, 2016*, pages 228–233, 2016.