DOCUMENT ANALYSIS OF MUSIC SCORE IMAGES WITH SELECTIONAL AUTO-ENCODERS

Francisco J. Castellanos[1] **Jorge Calvo-Zaragoza**[2]
Gabriel Vigliensoni[3] **Ichiro Fujinaga**[3]
[1] Software and Computing Systems, University of Alicante, Spain
[2] PRHLT Research Center, Universitat Politècnica de València, Spain
[3] Schulich School of Music, McGill University, Canada
fcastellanos@dlsi.ua.es

ABSTRACT

The document analysis of music score images is a key step in the development of successful Optical Music Recognition systems. The current state of the art considers the use of deep neural networks trained to classify every pixel of the image according to the image layer it belongs to. This process, however, involves a high computational cost that prevents its use in interactive machine learning scenarios. In this paper, we propose the use of a set of deep selectional auto-encoders, implemented as fully-convolutional networks, to perform image-to-image categorizations. This strategy retains the advantages of using deep neural networks, which have demonstrated their ability to perform this task, while dramatically increasing the efficiency by processing a large number of pixels in a single step. The results of an experiment performed with a set of high-resolution images taken from Medieval manuscripts successfully validate this approach, with a similar accuracy to that of the state of the art but with a computational time orders of magnitude smaller, making this approach appropriate for being used in interactive applications.

1. INTRODUCTION

The Optical Music Recognition (OMR) is a computational process that reads musical notation from images, with the aim of automatically exporting the content to a structured format [1]. Given the complexity of the task, the process is usually divided into different stages, the first of which is the document analysis. This stage consists of detecting and categorizing the different sources of information that appear in images of musical scores—e.g., classifying each pixel into one of four possible categories: background, staff line, musical note, or lyrics—and it is important for creating robust OMR systems [29]. That is, if subsequent stages receive the image in a reliable state, systems tend to generalize more easily.

Many researchers have proposed different algorithms to deal with specific steps within the document processing stage of the Optical Music Recognition (OMR) workflow. Traditionally, these strategies consist of heuristic workflows specifically designed for the scores at hand, exploiting specific details of the images to improve the performance of the detection. Music documents, however, especially from the Medieval and Renaissance era, come in a wide variety of notational styles and formats, resulting in a heterogeneous collection. Therefore, the previous approaches may be beneficial in the short term but they do not scale well [4, 6]. In many cases, a workflow must be developed anew for dealing with manuscripts with different notation, from a disparate time period, or with a differing level of image degradation.

Recent work has demonstrated the feasibility of using machine learning for document analysis [21, 25, 36]. In comparison to systems with hand-crafted heuristic rules, the advantage of using machine learning-based techniques lies in their generalizability, only needing labeled examples to build a new classification model [12]. In addition to this important advantage, the use of these techniques, in particular Convolutional Neural Networks (CNN), has proven to outperform the traditional strategies considered for document analysis in the OMR domain [6]. The main idea behind this approach is training a CNN to distinguish the category to which each pixel of the image belongs. That is, given a pixel of the image, and taking into account the pixels of its neighborhood, a model is trained to predict the category (e.g., note, staff line, and lyrics). In this way, the document analysis process consists of classifying every single pixel of the image into its actual category, thus separating the different layers of the document accordingly. Given that the classification is performed at pixel level, thin elements such as staff lines, note stems, as well as small artifacts, can be properly detected.

The problem with the aforementioned process is that it entails a high computational cost because it needs to classify every single pixel of an image. Since OMR is a process that lends itself to be used interactively [8, 9], there is a need of accelerating the processing of documents without sacrificing the classification quality, in order to present

a user-friendly environment.

We present in this paper a new framework based on machine learning that replaces the current pixel-wise model by a patch-wise model. In this approach, we process a complete sub-image (patch) in a single step, making predictions of many pixels simultaneously. This can be carried out by means of neural networks that learns how to compute an image-to-image prediction.

We evaluate the new approach over a set of high-resolution images taken from Medieval music manuscripts. The patch-wise model attains a similar accuracy to that of the state of the art but reducing the computational cost by several orders of magnitude.

The rest of the paper is organized as follows. We give a brief review of related work in Section 2. A formalization of the task, as well as the proposed solution, is detailed in Section 3. We empirically demonstrate in Section 4 that our model drastically reduces the computational time by orders of magnitude without the classification quality. Finally, we summarize the main conclusions of the present work in Section 5, pointing out some potential future work.

2. BACKGROUND

The classical workflow for OMR considers an initial document analysis stage [29], to process the input image before proceeding to the automatic recognition of the content. This first stage is crucial to increase the robustness of the system and to reduce the complexity of subsequent stages by providing correctly segmented images.

A common first step within the document analysis stage is binarization, in which background and foreground layers are separated. In addition to typical document image binarization techniques [15, 19, 30], some music-specific document binarization techniques have been proposed [28, 35]. Next, if the lyrics are part of the musical content, they need to be recognized as well. This is why there have been some proposals to separate the staves and the text [3, 7]. Once staff sections have been isolated, staff-line removal may take place. Although staff lines are necessary for music interpretation, most OMR workflows are based on detecting and removing the staff lines to perform connected component analysis on the remaining musical symbols. A comprehensive review and comparison of the first attempts for staff-line removal can be consulted in Dalitz et al. [10], and new techniques are being continuously developed [11, 13, 16]. In addition to these stages, we also find very specific processes that depend on the specific characteristics of the manuscript of interest, such as measure isolation [33], page-border removal [26], or frontispiece detection in Medieval manuscripts [31].

Recently, the full document processing of music score images has been implemented using CNNs, which learn to classify each pixel of the image according to its category [6]. This approach allows the analysis of entire documents with a generic method to any type of manuscript as long as there is appropriate training data. In addition to these advantages, this approach has proven to outperform the traditional strategies, and so it can be considered the state of the art in document analysis of music score images.

However, this process takes a long time because it has to perform an independent classification for each pixel of the image. Since images used are usually at high resolution involving millions of pixels, the resulting long computational time prevents its use in an interactive machine learning environment, where the user expects quick responses from the machine learning process while training it. Hence, in this work, we propose an image-to-image approach using neural networks, with the aim of maintaining the advantages of the state of the art but dramatically reducing the temporal cost.

3. FRAMEWORK

Formally, we define the task of document analysis of music score images as the process of assigning a category to each pixel of the image based on the layer of information to which it belongs. Specifically, we instantiate the task to the set of categories {`background`, `note`, `staff line`, `text`}. The reasoning behind this set is that it consist of the layers that lead to a general analysis of the image for the purpose of OMR, given that: musical notes are essential to recover the musical information; staff lines are necessary to divide the score into staves, as well as to estimate the pitch of the notes; text is also key for music interpretation but its information must be recognized with different algorithms (i.e., Optical Character Recognition); the rest of pixels can be considered as background. However, we show below that the chosen formulation can be extended to any other type of category set provided that sufficient labeled data is available.

As mentioned above, the aim of this work is to alleviate the computational cost involved in a pixel-wise classification approach. We address the issue here by using a set of auto-encoders, which learn an image-to-image mapping. Within our context, this means that the image can be processed in one step at a higher order of efficiency.

Conventional auto-encoders consist of feed-forward neural networks for which the input and output must be exactly the same. The network typically consists of two stages that learn the functions f and g, which are called encoder and decoder functions, respectively. Formally speaking, given an input x, the network must minimize a divergence $L(x, g(f(x)))$. An auto-encoder might initially appear to be pointless because it is trained to learn the identity function. Nevertheless, the encoder function f is typically forced to produce a representation with a lower dimensionality than the input. The encoder function therefore provides a meaningful compact representation of the input, which might be of great interest for feature learning or dimensionality reduction [37].

In our case, we modify this traditional behavior so that the model specializes in selecting the pixels that belong to each of the elements from the category set. This type of model is referred to as Selectional Auto-Encoder (SAE) [13]. An SAE is trained to perform a function such that $s : \mathbb{R}^{(w \times h)} \rightarrow [0, 1]^{(w \times h)}$. In other words, it learns a

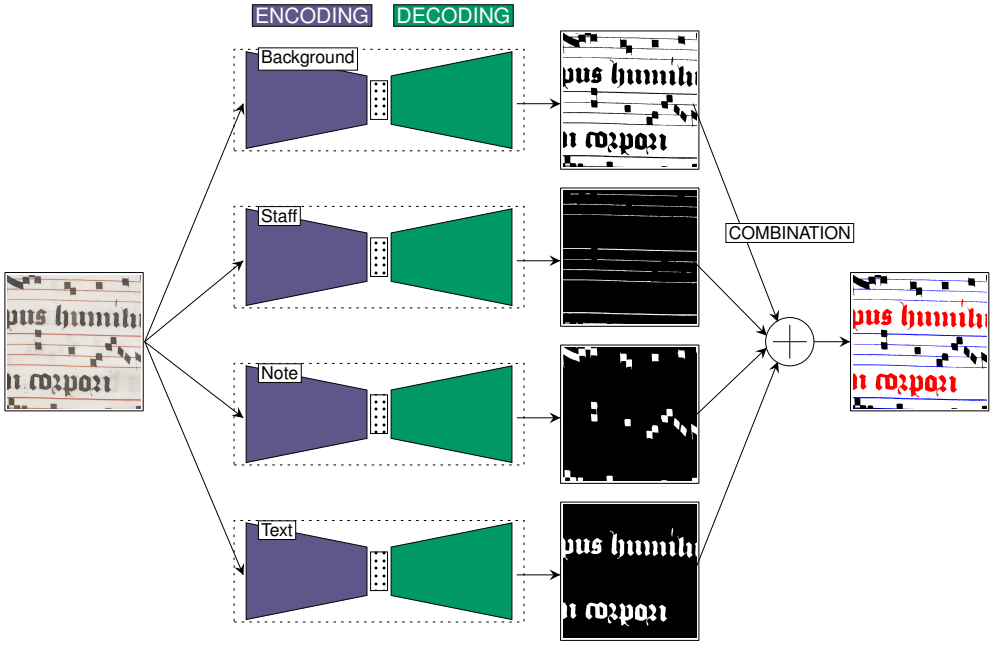

Figure 1. Graphical scheme of the SAE-based *1-vs-all* approach for document analysis of music scores images. The outputs of the individual SAE are represented as grayscale masks in which the white color represents the maximum selectional value. Coloring for the final combination: background in white, music symbols in black, staff lines in blue, and text in red.

binary map over a $w \times h$ image that preserves the input shape. The predicted value for each pixel indicates its selection level, representing 1 as the maximum. Then, the network is trained to minimize the divergence between a binary image in which only the pixels that belong to the category of interest are activated.

Actually, an SAE represents a two-class categorizer with one class represented by the value *0* and another represented by the value *1*. To perform a multi-class document analysis like the one formalized above, we follow a *1-vs-all* strategy, much in the same way as other binary classifiers such as the Support Vector Machine [20]. That is, we train a different SAE focused on each category, assuming the category of interest as 1 and the remaining ones as 0. At the time of inference, the outputs of all the trained SAEs are combined to obtain a global analysis of the document.

We find two important advantages of predicting each layer separately. On the one hand, the extraction of a specific layer only requires the ground-truth data of the targeted category, thus reducing the effort involved in preparing the training set if only a subset of the categories is pursued. On the other hand, the predictions provided by each SAE could be processed separately—e.g., to apply different thresholds to each result or to resolve inconsistencies when many predictions disagree about a specific region—which might be interesting depending on the way the subsequent stages of the OMR workflow operate.

Below we discuss more details about the actual implementation of the described framework for the present work.

3.1 Implementation details

An SAE can be configured in many ways. We specifically consider a Fully-Convolutional Network (FCN) topology, given the good results obtained by this type of neural networks in this task [32], and in general for any image-related task [23].

An FCN is a type of neural network that is entirely based on filters (i.e., convolutions). These filters are configured in a hierarchy of layers that provide multiple representations from the input image with different levels of abstraction: while the first layers emphasize details of the image, the last layers focus on high-level entities [22]. The parameters of the convolutions are typically optimized by backpropagation [24] through a training set, with the objective of generalizing to unseen data.

Consequently, the hierarchy of layers of our SAE consists of a series of convolutional plus pooling layers, until an intermediate layer is attained. As these layers are applied, filters are able to relate parts of the image that were initially far apart. Then, it follows a series of convolutional plus up-sampling layers that reconstruct the image up to the same input size copying neighboring pixels. The last layer consists of a set of neurons with *sigmoid* activation that predict a value in the range of $[0, 1]$, depending on the *selectional* level predicted for the corresponding input pixel. This selectional level is expected to approach 1 as the model is more confident that the pixel belongs to the category of interest. This specific configuration needs to be tweaked for the problem at issue, and so we will perform some preliminary experiments to evaluate different options.

The training stage consists of providing the SAE with

Corpus		Salzinnes	Einsiedeln
Pages		10	10
Avg. height and width per page (in pixels)		$5\,100 \times 3\,200$	$5\,550 \times 3\,650$
%	Background	80.6	79.1
	Note	11.2	10.0
	Staff line	4.5	6.9
	Text	3.7	4.0

Table 1. Overview of the corpus used in our experiments: number of pages, average size per page, and class distribution (in %).

examples of images and their corresponding ground truth, that is, binary maps over the pixels that belong to the category of interest. The *cross-entropy* loss function between each output activation and its expected activation is computed. Then, filters are tuned using stochastic gradient descent optimization [2] with a mini-batch size of 16 and the adaptive learning rate strategy proposed by Zeiler [38].

Once all the corresponding SAEs for the categories considered in this work ($SAE_{background}$, SAE_{note}, SAE_{staff}, SAE_{text}) are trained, they can be used to perform the document analysis process. In order to compute a single category for each pixel, we select the category whose SAE retrieves the highest selection value. A graphical scheme of this operation is depicted in Figure 1.

Given that our SAE is configured as a fully-convolutional model (i.e., without any dense layer), the input and the output layers can be of an arbitrary size. In practice, however, processing a high-resolution musical score has a high memory consumption. This is why in our case we need to divide the input music score into equal patches of 256×256 pixels, which was the largest size feasible with our computational resources. Theoretically, this limitation should not affect the performance of the models except for the case of the edges of the input patches. This can be palliated by considering overlap at the time of splitting the input image, and ignoring the edges of the predictions made.

4. EXPERIMENTS

4.1 Experimental setup

For the evaluation of our approach, we consider high-resolution image scans of two ancient music manuscripts. The first corpus is a subset of 10 pages of the Salzinnes Antiphonal manuscript (CDM-Hsmu M2149.14), [1] music score dated 1554–5. The second corpus is 10 pages of the Einsiedeln, Stiftsbibliothek, Codex 611(89), from 1314. [2] Table 1 gives an overview of this corpora with some of their specific features. For our experiments, the images have been considered in their grayscale format.

[1] https://cantus.simssa.ca/manuscript/133/

[2] http://www.e-codices.unifr.ch/en/sbe/0611/

The ground-truth data was created manually by labeling pixels into the four categories mentioned above. Although in this work we circumscribe the experiments to corpora from Medieval music manuscripts, we believe that their difficulty and wealth of information (at the image level) allows us to generalize the conclusions to any type of music score image.

In order to provide a more reliable assessment, we follow a corpus-wise 5-fold cross validation scheme. In each iteration of each corpus, 2 complete pages—not necessarily consecutive ones—are used for test evaluation, 2 pages are used as validation, and 6 pages for training the SAE models. The reported results will represent averages over these 5 independent evaluation processes. It be should noted that the experiments in both corpora have been performed individually, since in the context of machine learning, it could be assumed that the samples belong to the same domain. Despite this assumption, future research aims to expand the experimental setup to include more realistic scenario with cross-manuscript experiments.

As can be observed, the distribution of each class is highly biased, background being the most represented class. Given this distribution, we consider appropriate metrics for such imbalanced datasets. For instance, the F_1 typically represents a fair metric in these scenarios. In a two-class classification problem, this measure can be computed as

$$F_1 = \frac{2 \cdot \mathrm{TP}}{2 \cdot \mathrm{TP} + \mathrm{FP} + \mathrm{FN}}\,, \tag{1}$$

where True Positive (TP) stands for the correctly classified elements of the relevant class, False Positive (FP) represents the misclassified elements from the relevant class, and False Negative (FN) stands for the misclassified elements of non-relevant class.

To compute single values encompassing all possible categories, this metric can be reformulated into macro F_1 [27], which is computed as the average of all class-wise metrics.

4.2 Network selection

In this section we carry out a preliminary study to evaluate how some of the parameters of the SAE configuration affect the accuracy of the classification. It is worth mentioning that the different configurations may behave differently according to the category of interest (background, text, note, or staff). In this regard, however, we assume for this study a general assessment taking into account all classes simultaneously.

There exist a huge number of possibilities for establishing the organization of the neural model [18]. In order to reduce the search, we restrict ourselves to evaluate only the most interesting hyper-parameterization, namely the depth of the encoding/decoding blocks and whether encoding and decoding layer actually perform down-sampling and up-sampling operators. The latter points to an interesting issue: performing down- and up-sampling operations allows intermediate filters to focus on different levels

	Down/Up-Sampling	
Depth	No	Yes
1	90.0	90.7
2	91.5	94.9
3	93.3	96.0
4	94.2	95.4

Table 2. Macro average F_1 (%) of the 5-fold cross-validation over the validation partitions, with respect to the depth of the encoding/decoding layers and whether or not considering sampling operators.

of abstraction within the image, also reducing the intrinsic complexity—since the image in the intermediate layers would be smaller. However, keeping the original size throughout the process avoids having to learn to reconstruct the image, at the cost of losing the benefits discussed above for the opposite case.

The rest of the parameters are fixed manually, based on informal testing, as follows: the number of filters per convolution are set to 128 and the size of the convolutional kernels to 5×5. Also, all intermediate convolutional filters use Rectified Linear Unit (ReLU) activations [17].

Table 2 shows the macro average F_1 attained by each different SAE configuration on the validation sets.

Concerning the depth of the encoding/decoding blocks, a progress towards an upward trend is observed. In the case of using sampling operations, this trend finds a peak at 3 layers. In the opposite case (i.e., with no sampling), the improvements are more subtle and the peak is not reached within the number of layers considered. Due to computational resources, we were not able to carry out experiments with more layers, so it is not possible to know when the peak would be reached.

On the other hand, regardless of the number of layers chosen, we can observe that there is a clear tendency in the advantage of doing down- and up-sampling operations, since the latter case is always better than its analog for the same depth in the experiments carried out.

According to these results, the final SAE configuration for all the categories is shown in Table 3.

4.3 Results

In this section we analyze in detail the performance that was attained using the best SAE configuration of the previous section in comparison to the pixelwise CNN-based approach, that currently represents the state of the art in this task [5]. All experiments have been performed in similar conditions on a general-purpose computer with the following technical specifications: Intel(R) Core(TM) i7-7700HQ CPU @2.8GHz×4, 32GB RAM, GTX1070 GPU and Linux Mint 18.2 (64 bits) operating system. The code has been written using Python language (v2.7) and Keras framework.

Given that the objective of this paper is not only to measure the accuracy of the new model but also its efficiency, Table 5 shows a comparison of both aspects in terms of macro F_1 and the approximated time needed to process a document. Traditionally, the training cost is not taken into account when evaluating these systems because the process is usually performed offline. Note, however, that both approaches involved a similar training cost in the order of several hours on Graphical Processing Units.

Accuracy results show a visible difference between the corpora considered. While results are closer to the optimum in Salzinnes, both approaches seem to find more difficulties in Einsiedeln. However, this difference is not obvious in a qualitative evaluation, as depicted in Table 4.

It can be observed that the SAE-based strategy generally obtains a higher F_1 than that based on CNNs. Note, however, that the objective of this experiment is not to demonstrate that the SAE-based approach outperforms significatively the state of the art, but to obtain results that can be considered similar, which is clearly reported according to these figures. On the other hand, the computation time needed to process a complete manuscript page is drastically lower with the SAE, going from several hours to a few minutes. This happens because the CNN approach has to classify each pixel of the image, whereas the SAE approach can make predictions of many pixels simultaneously (in our experiments, 256×256). Obviously, the network of the latter approach is more complex, but it clearly compensates with respect to the temporal cost.

Thus, this comparison with the state of the art demonstrates that the proposed approach allows obtaining a similar performance when performing the document analysis, with a radically lower computational cost, thus making an important contribution to the field of OMR.

5. CONCLUSIONS

In this paper we have presented a machine-learning strategy for the document analysis of music score images. The strategy consists in training SAE, configured as convolutional neural networks, that allow to extract the different layers of information found in documents through an image-to-image formulation.

In a preliminary study, we have determined some of the parameters that lead to a better configuration of the SAE. In particular, we have evaluated the depth of the encoder/decoder layers, as well as the relevance of whether performing or not down- and up-sampling operations. Generally, increasing the number of layers is beneficial, to a certain extent, while sampling operators lead to a much more effective network.

Although we did not exhaustively test the various possible network configurations for this first study, we have shown that the proposed approach can achieve the accuracy similar to the state-of-the-art algorithms, and more importantly, with an efficiency improvement of orders of magnitude.

Our results represent the first step towards an interactive scenario in which the user and the system can interact to solve the OMR task. This scenario has already been devised before [34]; however, our approach allows us to be

Input	Encoding	Decoding	Output
$[0, 255]^{256\times256}$	Conv(128,5,5,ReLU)	Conv(128,5,5,ReLU)	$[0, 1]^{256\times256}$
	MaxPool(2,2)	UpSamp(2,2)	
	Conv(128,5,5,ReLU)	Conv(128,5,5,ReLU)	
	MaxPool(2,2)	UpSamp(2,2)	
	Conv(128,5,5,ReLU)	Conv(128,5,5,ReLU)	
	MaxPool(2,2)	UpSamp(2,2)	
		Conv(1,5,5,Sigmoid)	

Table 3. Detailed description of the selected SAE architecture, implemented as a FCN. Conv(f,h,w,a) stands for a convolution operator of f filters, with $h \times w$ pixel kernels with an a activation function; MaxPool(h,w) stands for the max-pooling operator with a $w \times h$ kernel and stride; UpSamp(h,w) denotes an up-sampling operator of h rows and w columns; *ReLU* and *Sigmoid* denote Rectifier Linear Unit and Sigmoid activations, respectively.

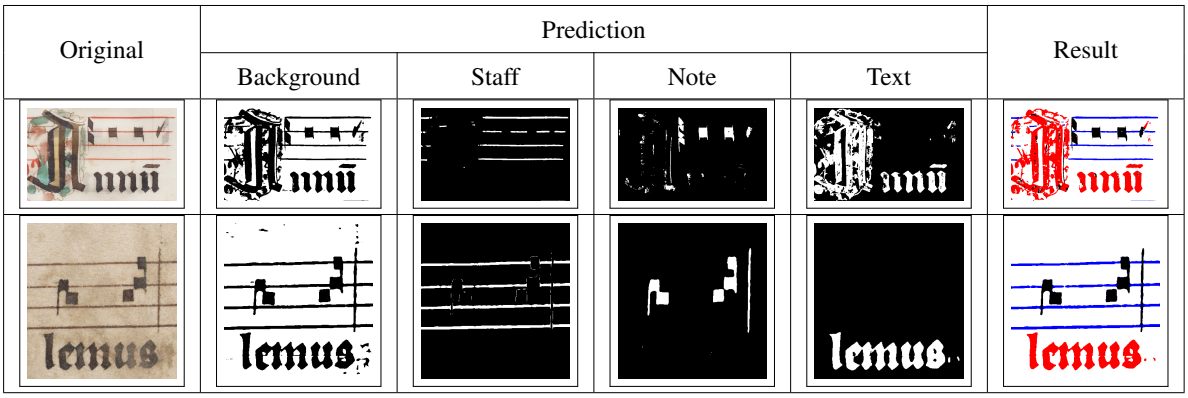

Table 4. Qualitative examples of document analysis over selected patches of the corpora (Salzinnes, first row; Einsiedeln, second row), depicting the original piece of the document along with the individual SAE predictions, and the resulting analysis. The predictions of the individual SAE are represented as grayscale masks in which the white color represents the maximum selectional value. Coloring for the final result: background in white, music symbols in black, staff lines in blue, and text in red.

Strategy	Macro F_1		Time per page
	Salzinnes	Einsiedeln	
SAE	95.5	90.3	$\sim$ 1 minute
CNN	91.3	88.4	$\sim$ 6 hours

Table 5. Comparison of our SAE-based approach with the state-of-the-art (CNN) performance taking into account both accuracy and efficiency of the document analysis process.

closer to real practice since the document analysis processing stage no longer implies a bottleneck.

Nevertheless, the costly training process is still an obstacle for this scenario in which models must re-trained according to user's corrections. Therefore, addressing this matter is essential in future work. Among the possible options, we want to consider the use of pre-trained models that can be adapted with few new samples and less demanding training procedures.

Also, we are especially interested in the aspect of cross-manuscript adaptation. That is, how to exploit models specifically trained for a manuscript in other manuscripts with a different layout organization. In this way, the initial effort to obtain ground-truth data from the manuscript at issue can be reduced. We believe that semi-supervised learning algorithms could be of interest in this case, for which the models learn to adapt to a new manuscript by just providing them with new (unlabeled) images. This can be performed by promoting convolutional filters that are both useful for the classification task and invariant with respect to the differences among manuscript types [14].

6. ACKNOWLEDGEMENT

This work was supported by the Spanish Ministerio de Economía, Industria y Competitividad through HispaMus project (TIN2017-86576-R) and Juan de la Cierva - Formación grant (Ref. FJCI-2016-27873), and the Social Sciences and Humanities Research Council of Canada.

7. REFERENCES

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

[2] L. Bottou. Large-scale machine learning with stochastic gradient descent. In *Proceedings of COMPSTAT'2010*, pages 177–186. Springer, 2010.

[3] J. A. Burgoyne 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 723–728, 2009.

[4] J. A. Burgoyne, L. Pugin, G. Eustace, and I. Fujinaga. A comparative survey of image binarisation algorithms for optical recognition on degraded musical sources. In *Proceedings of the 8th International Conference on Music Information Retrieval*, pages 509–512, 2007.

[5] J. Calvo-Zaragoza, F. J. Castellanos, G. Vigliensoni, and I. Fujinaga. Deep neural networks for document processing of music score images. *Applied Sciences*, 8(5):654–674, 2018.

[6] J. Calvo-Zaragoza, G. Vigliensoni, and I. Fujinaga. One-step detection of background, staff lines, and symbols in medieval music manuscripts with convolutional neural networks. In *Proceedings of the 18th International Society for Music Information Retrieval Conference, Suzhou, China*, pages 724–730, 2017.

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

[8] L. Chen and C. Raphael. Human-directed optical music recognition. *Electronic Imaging*, 2016(17):1–9, 2016.

[9] L. Chen, E. Stolterman, and C. Raphael. Human-interactive optical music recognition. In *ISMIR*, pages 647–653, 2016.

[10] 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.

[11] J. Dos Santos Cardoso, A. Capela, A. Rebelo, C. Guedes, and J. Pinto da Costa. Staff detection with stable paths. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31(6):1134–1139, 2009.

[12] R. O. Duda, P. E. Hart, and D. G. Stork. *Pattern Classification*. John Wiley & Sons, New York, NY, 2nd edition, 2001.

[13] A. Gallego and J. Calvo-Zaragoza. Staff-line removal with selectional auto-encoders. *Expert Systems with Applications*, 89:138–48, 2017.

[14] Y. Ganin and V. Lempitsky. Unsupervised domain adaptation by backpropagation. In *International Conference on Machine Learning*, pages 1180–1189, 2015.

[15] B. Gatos, I. Pratikakis, and S. J. Perantonis. Adaptive degraded document image binarization. *Pattern Recognition*, 39(3):317–327, 2006.

[16] T. Géraud. A morphological method for music score staff removal. In *International Conference on Image Processing*, pages 2599–2603, 2014.

[17] X. Glorot, A. Bordes, and Y. Bengio. Deep sparse rectifier neural networks. In *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics, Fort Lauderdale, FL*, pages 315–323, 2011.

[18] D. Graupe. *Principles of Artificial Neural Networks*. World Scientific Publishing Co., Inc., River Edge, NJ, USA, 2nd edition, 2007.

[19] N. R. Howe. Document binarization with automatic parameter tuning. *International Journal on Document Analysis and Recognition*, 16(3):247–258, 2013.

[20] C.-W. Hsu and C.-J. Lin. A comparison of methods for multiclass support vector machines. *IEEE Transactions on Neural Networks*, 13(2):415–425, 2002.

[21] F. D. Julca-Aguilar and N. S. T. Hirata. Image operator learning coupled with CNN classification and its application to staff line removal. In *14th IAPR International Conference on Document Analysis and Recognition, ICDAR 2017, Kyoto, Japan*, pages 53–58, 2017.

[22] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *26th Annual Conference on Neural Information Processing Systems*, pages 1106–1114, 2012.

[23] Y. LeCun, Y. Bengio, and G. Hinton. Deep learning. *Nature*, 521(7553):436–444, 2015.

[24] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

[25] B. Moysset, C. Kermorvant, C. Wolf, and J. Louradour. Paragraph text segmentation into lines with recurrent neural networks. In *13th International Conference on Document Analysis and Recognition*, pages 456–460. IEEE, 2015.

[26] Y. Ouyang, J. A. Burgoyne, L. Pugin, and I. Fujinaga. A robust border detection algorithm with application to medieval music manuscripts. In *Proceedings of the 2009 International Computer Music Conference*, pages 101–104, 2009.

[27] A. Özgür, L. Özgür, and T. Güngör. Text categorization with class-based and corpus-based keyword selection. In *International Symposium on Computer and Information Sciences*, pages 606–615, 2005.

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

[29] 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.

[30] J. Sauvola and M. Pietikäinen. Adaptive document image binarization. *Pattern Recognition*, 33(2):225–236, 2000.

[31] C. Segura, I. Barbancho, L. J. Tardón, and A. M. Barbancho. Automatic search and delimitation of frontispieces in ancient scores. In *18th European Signal Processing Conference*, pages 254–258, 2010.

[32] E. Shelhamer, J. Long, and T. Darrell. Fully convolutional networks for semantic segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(4):640–651, 2017.

[33] 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*, pages 125–130, 2013.

[34] G. Vigliensoni, J. Calvo-Zaragoza, and I. Fujinaga. An environment for machine pedagogy: Learning how to teach computers to read music. In *Proceedings of IUI Workshop on Music Interfaces for Listening and Creation, Tokyo, Japan*, pages 1–4, 2018.

[35] Q. N. Vo, S. H. Kim, H. J. Yang, and G. Lee. An MRF model for binarization of music scores with complex background. *Pattern Recognition Letters*, 69(Supplement C):88–95, 2016.

[36] Q. N. Vo, S. H. Kim, H. J. Yang, and G. Lee. Binarization of degraded document images based on hierarchical deep supervised network. *Pattern Recognition*, 74:568–586, 2018.

[37] W. Wang, Y. Huang, Y. Wang, and L. Wang. Generalized autoencoder: A neural network framework for dimensionality reduction. In *Workshops of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 490–497, June 2014.

[38] M. D. Zeiler. ADADELTA: An adaptive learning rate method. *Computer Research Repository*, abs/1212.5701, 2012.