Automatic page classification in a large collection of manuscripts based on the International Image Interoperability Framework

Emanuela Boroş*, Alexis Toumi†, Erwan Rouchet*,
Bastien Abadie*, Dominique Stutzmann‡, Christopher Kermorvant*§
*TEKLIA SAS, Paris, France
‡IRHT CNRS, Paris, France
†Department of Computer Science, University of Oxford, England
§LITIS, Université de Rouen-Normandie, France

***Abstract*—In patrimonial institutions such as libraries and archives, the valorization of the vast amount of documents that have been recently digitized is still a challenge. Most of these documents are freely accessible as images but their textual content remains largely unreachable and unknown. Research projects dedicated to specific collection allow creating meta-data or even transcriptions obtained through volunteers or crowd-sourcing. But the vast majority of the documents cannot be manually transcribed or indexed: automatic large-scale processes for indexing are needed. The increasing adoption of the International Image Interoperability Framework (IIIF) by the patrimonial institutions is a technological enabler for the development of such services. Images are accessible with a unique protocol across institutions and both images and data can be presented with standard tools. In this paper, we describe an architecture for automatic processing of historical documents owned by different institutions but processed and presented thanks to the IIIF framework. We implemented this architecture and processed a large collection of books of hours with a page classifier trained on an annotated sample. The result is freely distributed and can be viewed with any IIIF compatible viewer.**

***Index Terms*—Image classification, Machine learning, Feature extraction, Neural networks, Historical documents, IIIF.**

I. Introduction

The amount of handwritten documents from the past centuries preserved in public or private libraries and archives is tremendous. Ambitious digitization campaigns have been funded in recent years to protect this cultural heritage. Today, less than 1% of the handwritten documents have been digitized, most of these documents are freely accessible as images but their textual content remains largely unreachable and unknown. Despite the great progress in automatic document analysis and recognition achieved recently thanks to the breakthrough in machine learning and particularly deep learning methods, recognition technologies are not widely used on handwritten documents or early printed documents. Curators and librarians still think that these technologies are complex to enable and that their output is erratic and difficult to showcase. Regarding the added value to the users, recent success in the indexing of large collections of handwritten documents [1], [2] should change these misconceptions and encourage the adoption of these technologies. Regarding the integration of recognition technologies in existing infrastructure, large scale experiments on publicly available resources must be conducted to demonstrate its feasibility and simplicity.

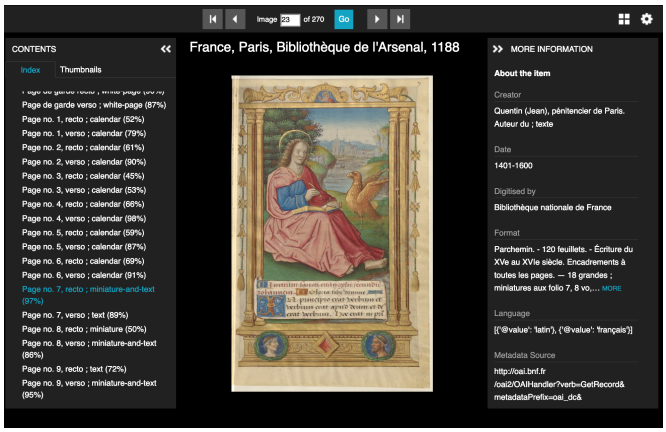

Fig. 1. Visualization of the page classification results in the UniversalViewer interface: the predicted page classes allow to quickly navigate to the pages of interest.

The goal of this paper is to demonstrate that document classification can be easily integrated and exploited on a large scale collection of manuscripts hosted by different institutions by leveraging the International Image Interoperability Framework (IIIF). We developed a page classification system based on deep learning algorithms and have integrated it in a back-end system allowing the access to images of the manuscripts located in different institutions in different countries and present the classification results using the standard IIIF visualization interfaces such as Mirador[1] or Universal Viewer[2]. Our contributions are the following:

- a description of an integrated system using automatic page classification and the IIIF framework for accessing

[1]http://projectmirador.org/
[2]http://universalviewer.io/

the images and presenting the results of processing a large collection of manuscripts
- the introduction of a collection of books of hours with annotated page classes
- a comparison of two deep learning-based historical page classifiers, with and without transfer learning, and a well-established machine learning-based baseline

II. Related work

First, we present in this section the different approaches that have been chosen to design interoperable document processing frameworks. We show that one of the main hindrances to their usage on very large corpora is the way images are accessed. Then we quickly review page classification techniques in the context of historical document processing. Finally, we describe the IIIF framework.

A. Document processing framework

Until recently, most of the research projects on historical document processing have developed specific user interfaces and back-end architectures tailored to their needs [1], [3], [4].

The European IMPACT project aimed to significantly improve access to historical text by developing OCR and language processing technologies[3]. Some of the services and resources continue to be provided by each member of the Impact Centre of Competence, but they are not available through a unified interface nor a single provider.

The DAE platform [5] was developed to provide an open architecture allowing to perform end-to-end document analysis benchmarking. It was designed in a fully modular way so that it was very easy to plug in and test new algorithms. The goal was rather to allow reproducible research than to provide transcription services. However, after several years, Lamiroy [6] concluded that the centralized data repository hindered the potential users to make use of the platform because they prefer hosting their data on their own server instead of giving it up to a third party. Following this trend, DIVAServices [7] was developed to provide easy access to document processing algorithms using RESTful web API but each image still has to be uploaded to the processing server.

Transkribus[4] is a research infrastructure for transcribing, recognizing and searching archival documents developed in the framework of the READ-H2020 e-Infrastructure Project. This platform is now the most widely used platform for transcription projects. An expert client GUI has been developed and if simple operations can be done locally, all automatic services are available only for remote documents, which are stored on the servers of the University of Innsbruck. This architecture is well adapted for the processing of a relatively small corpus of documents that can be easily uploaded to the platform. However, it is not suited for a corpus of documents distributed in different institutions and representing hundreds of thousands of pages.

[3]https://www.digitisation.eu

[4]https://transkribus.eu/

B. Historical document page classification

In order to index large collections of documents, several image processing can be considered, automatic page classification, the recognition of the kind of document (document classification), the separation of textual and graphical components in a document image (document analysis), the identification of semantically relevant components of the page layout (document understanding), the extraction of text from portions of the image document (OCR). Page classification is usually the first step in a document analysis system since it allows to reveal the structure of the collection and to present it to the user.

Currently, page classification can be grouped into image-based, content-based or a combination of both. Features extracted from document images can either be visual, textual or a combination of the two. Other features can be the percentage of text and non-text elements in a content region of image, font sizes [8], table structures, document structures [9], bag-of-words, and statistics of features are only a few examples of extracted combined textual and visual characteristics adopted by some of the previously cited works for solving the task of document image classification [10]–[12]. The authors of [13], [14] proposed a method based on low-level features (texture, shape, and geometric descriptors) to classify pages in historical documents. The same approach, but without looking for or taking into account the a priori knowledge of the structure of the pages, is described in [15].

Deep architectures based on Convolutional Neural Networks (CNNs) are well-known in the domain of object recognition and image classification. More recently [16] showed a great improvement in the accuracy by applying deeper models and transfer learning from the domain of real-world images to the domain of document images, thus making it possible to use deep CNN architectures even with limited training data. CNNs have been successfully applied to page classification in modern documents [17], [18].

C. The IIIF framework

The International Image Interoperability Framework was developed to facilitate multiple usages of images on the web, having recognized that most images are locked up in their primary application. It is implemented as an internet protocol, technically a JSON REST API. The IIIF protocol is implemented by IIIF servers (hosting images, metadata, and organization) and IIIF viewers or clients (who connect to the servers and display the content available). A set of five distinct REST APIs is used by both parts:

- *Image API* handles raw images descriptions and operations. A client can easily get a different version of an image (thumbnails, crops, with filters, etc.) by adding parameters to the Image URL;
- *Presentation API* is used to add hierarchical layers to the images, so that API clients can display them in a comprehensible way, using collections, sequences and pages layers for example;

- *Content search API* allows users to make search queries on the images metadata and text transcriptions;
- *A/V API* delivers time-based media (audio, video);
- *Authentication API* manages access rights to the four previously described APIs. It is entirely optional, as most of the available content is in the public domain;

These APIs are really simple to use, most of the operations are made using the HTTP GET method, so developers can test a server's capabilities by altering URLs parameters.

One interesting feature of this set of APIs, is their relative independence across servers: one image provider can give access to its images repositories and organization using the Image and Presentation API, but another party could provide a Search index, or even a different Presentation for the initial images, while relying on the initial Image API Server.

Thanks to its decentralized and interoperable design, we think that IIIF offers a real opportunity to develop automatic processing and enrichment of historical digital document collection kept in cultural institutions.

III. HORAE: AN INTERNATIONAL COLLECTION OF BOOK OF HOURS

A. The HORAE project

Books of hours form, with more than 10,000 preserved manuscripts, a vast and crucial ensemble to understand the medieval mindset. Yet, their textual content is very scarcely studied, although the massive production of such a large number of manuscripts is a pivotal cultural and industrial phenomenon and witness to the profound changes in late medieval society on cultural, religious, and industrial levels: speculative book production rather than on commission for specific clients, internalization of faith (*devotio moderna*) and imitation of clerical practices by lay people, customization of devotional objects, etc. Books of hours are at once deluxe items of social display, and intimate objects of devotional intensity, used for one's salvation. Books of hours have been too scarcely studied until now because they are too numerous, too complex and their text is very repetitive. They are also preserved in many different libraries across the world so that an automatic large scale analysis was but feasible before the adoption of the IIIF framework.

HORAE is a cross-disciplinary research project studying religious practices and experiences in the late Middle Ages through the books of hours, the absolute medieval best seller. The project aims at identifying manuscript clusters, which share the same textual characteristics in the order of the different parts (Officium Beatae Mariae Virginis, votive offices, suffrages, prayers), but also in the order of textual units to identify the liturgical use. This will allow studying the diffusion and circulation of devotional and liturgical texts at the end of the Middle Ages in order to better understand the cultures and faith in the 13th c.-16th c.

To reach this goal, each book of hours must be analyzed to extract its structure and the first step consists in classifying the pages into classes that reveal the structure of the manuscript. This step is presented in this paper.

B. Datasets

We established a page classification ground truth for 122 books of hours, for a total of 37,984 images. Among these, 86 manuscripts with a total of 28,744 pages are published by the BNF (Bibliothèque Nationale de France) and the remaining 36 come from the Médiathèque François Mitterrand/Espace Mendes France from Poitiers, and are called *Pictavenses* for short. The amount of 37,984 images are utilized for training, evaluation, and testing our proposed models. We also gathered 1000 images from manuscripts published by the Harvard Library[5], for testing purpose only.

	train and validation		test	
	Pictavenses	BNF	Harvard	total
binding	232	564	20	1233
white page	399	1719	42	2735
calendar	621	1458	61	2079
miniature	48	467	12	558
miniature and text	383	1355	32	1977
text with miniature	295	543	19	945
full-page text	7262	22638	814	35797
total	9240	28744	1000	38984

TABLE I
CLASS DISTRIBUTION

Due to the style of layout of the pages of the book of hours, we decided on the names of classes beforehand according to the page information needed for a historical or paleographical study: *binding, white page, calendar, miniature, miniature and text, text with miniature* and *full-page text*. The difference between *miniature and text* and *text with miniature* is in the size of the miniature which in the first case overwhelms the textual segment in the image.

In order to collect the samples per classes, we first applied a *KMeans* clustering technique [19] on Histogram of Oriented Gradients (HOG) features extracted from images of fixed size. HOG feature descriptors remain one of the few options for object detection and localization that can remotely compete with the recent successes of deep neural networks. HOG descriptors can capture outline information of text lines or miniatures and are considerably simpler, and faster alternatives to neural networks. First, we resize the images to 300×400 and extract patches of size 100 from the images, and for every patch, the returned features encode local shape information from the region. For each patch, we accumulate a local 1-D histogram of gradient over all the pixels in the patch. Each orientation histogram divides the gradient angle range into an 8 number of bins. We obtain feature vectors of size 192 once flattened. We also extract a 3D RGB color histogram with 8 bins per channel, yielding 24-dimensional feature vectors. These features are concatenated with a final size of 261 and passed to the clustering algorithm. We manually found the best value for the number of clusters and then cleaned the clustering by moving images to their correct cluster. This manual process

[5]https://library.harvard.edu/

took two days for one person. The resulting class distribution is presented in Table I. The dataset is publicly available[6].

IV. Automatic Page Classification

We first classify the pages of our corpus into seven classes: on one side, we filter out pages without text (*bindings, white pages*, and miniatures), on the other, we want to classify textual pages according to their layout (*full-page text, large miniatures with text*, or *mostly text with small miniatures*, and *calendars*).

A. Pre-processing

For all the models used in this paper, we use images of a fixed size as input, we downscale all images to 224×224. After re-sizing the images, we compute the mean pixel values of the training images and subtract them from all images to center the training data. As a last preprocessing step, we convert the grayscale images to RGB images by simply copying the pixel values of the single-channel images into three channels. We split our ground truth (shown in Table I) into a training set of 97 manifests (29816 images) and 25 manifests (8168 images) for testing. Splitting on the manifests rather than randomly sampling images allows us to give a better measure of generalization for unseen manuscripts. As an internal evaluation, we used 20% of the training set. In order to evaluate the robustness of our models, we also test their performance on 1000 random sampled images provided by Harvard Library.

B. Baseline model

Since the interest is in finding images with full-page text, or text with miniatures, we chose as a baseline, a Random Forest with 100 estimators applied also on the extracted Histogram of Oriented Gradients (HOG) features and color histograms. The features are extracted in the same manner presented in Section III-B.

	precision	recall	f1-score	support
binding	0.85	0.86	0.85	173
white page	0.93	0.93	0.93	399
calendar	0.34	0.32	0.33	380
miniature	0.33	0.19	0.24	107
miniature and text	0.74	0.69	0.71	411
text with miniature	0.38	0.03	0.06	198
full-page text	0.93	0.96	0.94	6500
meaningful text	0.59	0.50	0.51	7489

TABLE II
Classification report BnF & *Pictavenses* for the baseline model based on Histogram of Oriented Gradients (HOG) features and a Random Forest classifier

We obtain an overall accuracy score of 87.94% (±0.25) – an average on 5 runs –, shown in comparison with other models in Table IV. If we restrict the classification task to deciding whether the page contains meaningful text (i.e. its class is *full-page text, text with miniature* or *miniature and text*) then we obtain a 51% F1-score. Detailed results are presented in Table II.

[6]https://github.com/oriflamms/HORAE/

C. Convolutional neural network-based model

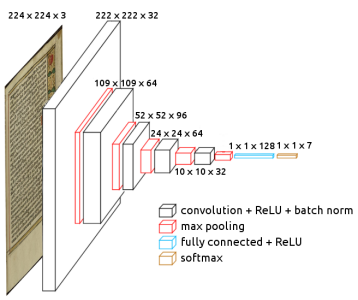

Fig. 2. CNN architecture for page classification. Figure inspired by [20]

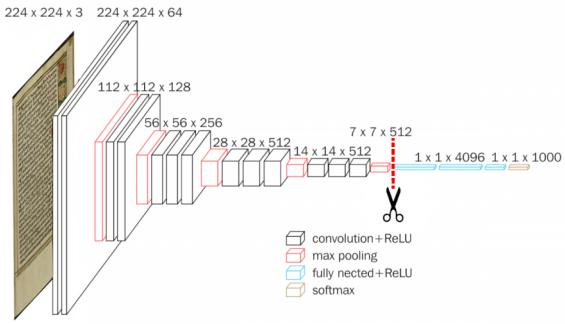

Fig. 3. VGG-16 architecture for page classification. Figure from [20]

To move beyond the baseline model, we implemented a deep convolutional neural network, designed to classify the pages. This network is composed of six layers. Five convolutional layers with (32, 64, 96, 64, 32) kernels of size 3 and a stride of 1 pixel. A max pooling with size 2 and a stride of 1 pixel is applied after all convolutional layers. These are followed by one fully connected layer of size 128. The output of the last fully-connected layer is fed to a 7-way *softmax* which produces a distribution over the class labels (*binding, white page, calendar, miniature, miniature and text, text with miniature*, and *full-page text*). A *ReLU* non-linearity and a batch normalization layer that normalizes each input channel across a mini-batch, are applied after every convolutional or fully connected layer, in order to speed up the training and to reduce the sensitivity to network initialization. The architecture is summarized in Figure 2 and trained using *Adam* optimizer [21], batch size of 16, with *He* weight initialization [22] for the convolutional and the fully connected layer of size 128, with a total of trainable parameters of 252,228. We also use early stopping as a form of regularization used to avoid overfitting, with a patience of 2 epochs, and a dropout of 0.4 after the dense layer before prediction.

We obtain an overall accuracy score of 94.65% (±0.55) – an average on 5 runs – , as well as an average F1-score of 95% (see Table IV that reflects the best run). If we restrict the classification task to deciding whether the page contains meaningful text (i.e. its class is *full-page text, text*

with miniature or *miniature and text*) then we obtain an 83% F1-score.

D. A very deep convolutional network for transfer learning

We also tested an off-the-shelf pre-trained model as a feature extractor, VGG-16, described in [20]. For these experiments, the optimizer, batch size, initialization, and regularization are the same in our proposed model.

The VGG network architecture was introduced by [20], in 2014, and it was originally trained for the purpose of object classification. The model achieved a top-5 test accuracy in *ImageNet*, which is a dataset of over 14 million images belonging to 1000 classes. It was one of the famous model submitted to ImageNet Large-Scale Visual Recognition Challenge (ILSVRC)[7], in 2014. The pre-trained weights of VGG trained on *ImageNet* [23] are available for fine-tuning. This network is characterized by its simplicity, using only 3×3 convolutional layers stacked on top of each other in increasing depth. Reducing the volume size is handled by max pooling. Two fully-connected layers, each of size 4096 are then followed by a $softmax$ layer, as shown in Figure 3. The 16 and 19 stand for the number of weight layers in the network. The key of transfer learning is to just leverage the pre-trained model's weighted layers to extract features but not to update the weights of the model's layers during training with new data for the new task.

	precision	recall	f1-score	support
binding	0.97	0.76	0.85	173
white page	0.90	0.98	0.86	399
calendar	0.89	0.84	0.86	380
miniature	0.44	0.67	0.53	107
miniature and text	0.88	0.79	0.83	411
text with miniature	0.73	0.64	0.68	198
full-page text	0.98	0.99	0.98	6500
meaningful text	0.87	0.81	0.83	7489

TABLE III
CLASSIFICATION REPORT BNF & *Pictavenses* FOR THE VGG-16 WITHOUT THE TWO FULLY CONNECTED LAYERS BEFORE PREDICTION

We tested the VGG-16 pre-trained network without the two fully-connected 4096-dimensional, as illustrated in Figure 3. This network took much longer to train than our model, even though it has a total of trainable parameters of 175,623 (and 14M non-trainable parameters), due to the calculation of the features extracted by the convolutional layers. The results are presented in more detail in III (the best run between the averaged results presented in Table IV).

Since it is quite common to also fine-tune the two fully-connected 4096-dimensional layers of VGG-16 [17], [24], [25], we experimented it on our task, but it proved to be too complex for the amount of data available and it resulted in detecting every image as being a *full-page text image*, and thus a macro average of the scores per class revealed a very low F1-score of 13%. We excluded the detailed view of the results per class from this work, since all of the F1-scores are 0%, except for the *full-page text*, which is 89%.

[7]http://www.image-net.org/challenges/LSVRC/2014/results

Model	Accuracy %	
	BnF & *Pictavenses* (average on 5 runs)	Harvard samples
Baseline HOG	87.94 ($\pm$0.25)	65.70
Baseline CNN	94.65 ($\pm$0.55)	86.20
VGG16 (pre-trained, with the two dense layers)	79.39 ($\pm$0.37)	81.40
VGG16 (pre-trained, without the two dense layers)	**94.92 ($\pm$0.12)**	**92.10**

TABLE IV
ACCURACY RESULTS ON BNF & *Pictavenses* FOR DIFFERENT MODELS, AVERAGED ON 5 RUNS

Model	train	test	test (per sample)	epochs %
Baseline HOG	3.2 s	2 ms	$2\mu s$	–
Baseline CNN	226 s	10 s	1 ms	4
VGG16 (pre-trained, with the two dense layers)	833 s	34 s	4 ms	3
VGG16 (pre-trained, without the two dense layers)	702 s	33 s	4 ms	6

TABLE V
PROCESSING TIME REPORT FOR TRAINING AND TESTING

From Table IV, we can conclude that our proposed CNN architecture has similar results with the VGG-16 without the two fully-connected layers before prediction, and it takes considerably much less time than using transfer learning with a VGG-16, as shown in Table V, and thus it is appropriate for this task. At the same time, we are aware that the amount of data, 45,511 images, might not be enough for a good generalization, and since we do not enlarge our training dataset in any way but train solely with images containing the entire document, a data augmentation technique may be considered in the future.

	precision	recall	f1-score	support
binding	0.29	0.90	0.43	20
white page	0.58	1.00	0.73	42
calendar	0.64	0.48	0.55	61
miniature	0.50	0.25	0.33	12
miniature and text	0.60	0.97	0.74	32
text with miniature	0.73	0.42	0.53	19
full-page text	0.97	0.90	0.93	814
meaningful text	0.73	0.82	0.68	926

TABLE VI
CLASSIFICATION REPORT ON THE HARVARD SAMPLES

We tested all models on 1000 randomly sampled images provided by Harvard Library. As one can see in the Table IV, we obtain an accuracy of 86.20% with our proposed CNN, which can only mean that Baseline CNN lacks the ability of generalization, and thus we can draw the same conclusion as previously stated, a type of data augmentation may be more effective. A more detailed view of the results of our proposed CNN is presented in Table IV, where one can observe the F1-score for the meaningful text of 73%. In the case of VGG-16 with the two fully connected layers fine-tuned, we obtain an accuracy of 81.40%, but this is due to the fact that all the pages have been classified as *full-page text* images. The baseline model does not generalize either, and thus the best performing

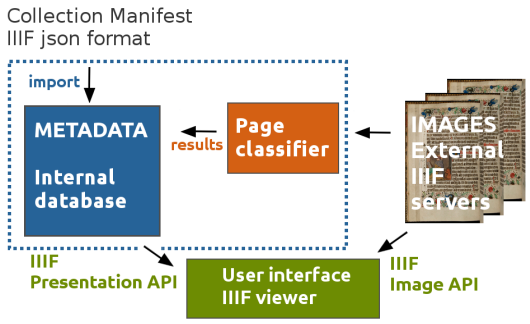

Fig. 4. The HORAE document processing architecture based on the IIIF APIs

model is the VGG-16 without the fully connected layers, and only the weights of the prediction layer to be learned. This proves that the feature extraction provided by the pre-trained convolutional layers is to be considered as a reliable solution for the historical page classification task.

E. HORAE, a document processing architecture based on IIIF

The HORAE project uses extensively the IIIF APIs described above, specifically the Image API and Presentation API. Every image used in the project, from full-size images for classification to thumbnails used in the frontend displays, is served by IIIF servers. We support a wide range of servers from organizations across the world, allowing us to use their library of images to apply our classification algorithms and display the results in a user-friendly way. This architecture is shown in Figure 4.

The backend application maintains images lists from those servers, and allows to organize them in hierarchical categories (books, volumes, corpus, etc.) and generates dynamically new IIIF manifests that are compatible with common visualization tools (Mirador, Universal Viewer). All the manuscripts and the classification results presented in this paper are freely accessible at https://arkindex.teklia.com/.

V. Results and Visualization

We used the proposed architecture and CNN-based classification model to process a large collection of five hundreds book of hours, representing 105,514 pages and hosted in different institutions. The IIIF manifest of this collection is freely available[8].

The full processing took around 10 days, which corresponds to about 8 seconds per page: 77% for image access, 22% for page classification and 1% for data storage.

Once the class of each page in the manuscripts have been predicted and stored, they are added to IIIF presentation manifest and can be visualized with any IIIF compatible viewer, as shown on Figure 1.

Some unsupervised metric can be computed to monitor the behavior of the classifier over the different manuscripts and detect candidates for manual correction or new annotations. A first unsupervised metric is the average classification score per manuscript. The distribution of this score is shown in figure 5 for all the manuscripts of the collection, together with a sample of pages. Manuscripts with the lowest average classification score are more likely to have bad classification results. This was confirmed by manual visualization of the predicted page classes. Manuscripts with double page digitization tend to have lower score whereas grayscale images seem not to be problematic. Another metric is the predicted page class distribution within a given manuscript. This distribution should be similar to that of the annotated set. For each manuscript, the Kullback-Leibler (KL) distance between the predicted and expected class distribution can be computed. This metric can be combined with the previous one to spot problematic manuscripts as shown in Figure 6. Manuscripts with a low KL distance whereas high average score are more likely to be well predicted and manuscripts with high KL distance and the low average score should be examined.

[8]https://github.com/oriflamms/HORAE/

VI. Conclusion and Future Work

In this paper, we have presented the architecture of a document processing workflow based on the IIIF framework. We believe that the adoption of the IIIF framework in cultural institutions worldwide can really promote the use of automatic processing of the historical document and develop new usages. We implemented this architecture and applied automatic page classification to a large collection of books of hours hosted in different institutions. The results are presented in a uniform way and accessible with any IIIF compatible viewer. This proof of concept raised many questions that we will tackle in the future.

First, we processed only 500 manuscripts, but there are probably more than 1000 books of hours and tens of thousands of medieval manuscripts provided in IIIF format. For all of them, automatic page classification would enhance the access and use and prepare further automated processes. We will increase the HORAE dataset as we find more books of hours. Second, only the first stage of document processing has been implemented so far: page classification. This step is the quickest one and new problems will arise when we will be dealing with document layout analysis and handwritten text recognition. Third, we need to develop efficient unsupervised quality measures, to be able to detect manuscripts that are defectively processed so that we can annotate some of their pages, retrain the model and reprocess them. Finally, we should develop ways to take into account users' feedback and define strategies to know when to retrain or update the models and on which annotated the data.

References

[1] T. Bluche, S. Hamel, C. Kermorvant, J. Puigcerver, D. Stutzmann, A. H. Toselli, and E. Vidal, "Preparatory KWS Experiments for Large-Scale Indexing of a Vast Medieval Manuscript Collection in the HIMANIS Project," *International Conference on Document Analysis and Recognition*, 2017.

[2] F. Bolelli, G. Borghi, and C. Grana, "Xdocs: an application to index historical documents," in *Italian Research Conference on Digital Libraries*. Springer, 2018, pp. 151–162.

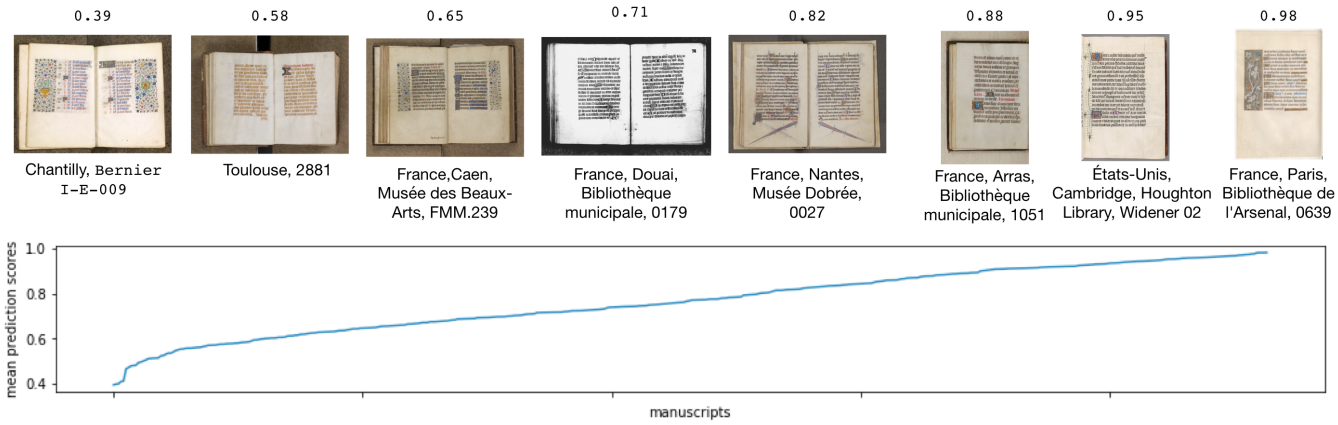

Fig. 5. Distribution of the average classification score per manuscript and some corresponding examples

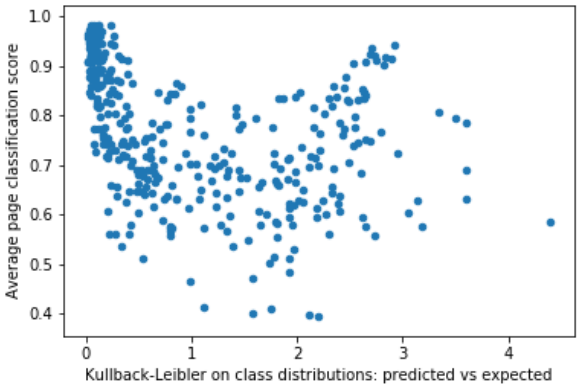

Fig. 6. Distribution of the manuscripts according to two unsupervised quality metrics : average class score and the Kullback-Leibler distance between the predicted and expected class distribution.

[3] T. Causer and V. Wallace, "Building A Volunteer Community: Results and Findings from Transcribe Bentham," *Digital Humanities Quarterly*, vol. 6, 2012.

[4] L. Schomaker, "Design considerations for a large-scale image-based text search engine in historical manuscript collections," *Information Technology*, vol. 58, no. 2, 2016.

[5] B. Lamiroy and D. Lopresti, "An Open Architecture for End-to-End Document Analysis Benchmarking," in *International Conference on Document Analysis and Recognition*, 2011.

[6] B. Lamiroy, "DAE-NG: A Shareable and Open Document Image Annotation Data Framework," in *International Conference on Document Analysis and Recognition*, 2018.

[7] M. Würsch, R. Ingold, and M. Liwicki, "SDK Reinvented: Document Image Analysis Methods as RESTful Web Services," in *International Workshop on Document Analysis Systems*, 2016.

[8] C. Shin, D. Doermann, and A. Rosenfeld, "Classification of document pages using structure-based features," *International Journal on Document Analysis and Recognition*, vol. 3, no. 4, pp. 232–247, 2001.

[9] F. Cesarini, M. Lastri, S. Marinai, and G. Soda, "Encoding of modified xy trees for document classification," in *International Conference on Document Analysis and Recognition*, 2001, pp. 1131–1136.

[10] C. K. Shin and D. S. Doermann, "Document image retrieval based on layout structural similarity," in *Conference on Image Processing, Computer Vision, Pattern Recognition*, 2006, pp. 606–612.

[11] J. Kumar and D. Doermann, "Unsupervised classification of structurally similar document images," in *International Conference Document Analysis and Recognition*, 2013, pp. 1225–1229.

[12] N. Chen and D. Blostein, "A survey of document image classification: problem statement, classifier architecture and performance evaluation," *International Journal of Document Analysis and Recognition*, vol. 10, no. 1, pp. 1–16, 2007.

[13] M. Mehri, P. Heroux, J. Lerouge, P. Gomez-Kramer, and R. Mullot, "A structural signature based on texture for digitized historical book page categorization," in *International Conference on Document Analysis and Recognition*, 2015.

[14] M. Mehri, P. Héroux, J. Lerouge, and R. Mullot, "Page retrieval system in digitized historical books based on error-tolerant subgraph matching," in *International Conference on Document Analysis and Recognition*, vol. 1, 2017, pp. 1168–1173.

[15] N. Journet, J.-Y. Ramel, R. Mullot, and V. Eglin, "Document image characterization using a multiresolution analysis of the texture: application to old documents," *International Journal of Document Analysis and Recognitiong*, vol. 11, no. 1, pp. 9–18, 2008.

[16] M. Z. Afzal, A. Kölsch, S. Ahmed, and M. Liwicki, "Cutting the error by half: Investigation of very deep cnn and advanced training strategies for document image classification," in *International Conference on Document Analysis and Recognition*, vol. 1, 2017, pp. 883–888.

[17] C. Tensmeyer and T. Martinez, "Analysis of convolutional neural networks for document image classification," in *International Conference on Document Analysis and Recognition*, 2017.

[18] A. Kölsch, M. Z. Afzal, M. Ebbecke, and M. Liwicki, "Real-Time Document Image Classification using Deep CNN and Extreme Learning Machines," in *International Conference of Document Analysis and Recognition*, 2017.

[19] J. MacQueen *et al.*, "Some methods for classification and analysis of multivariate observations," in *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*, vol. 1, no. 14. Oakland, CA, USA, 1967, pp. 281–297.

[20] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[21] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[22] K. He, X. Zhang, S. Ren, and J. Sun, "Delving deep into rectifiers: Surpassing human-level performance on imagenet classification," in *International Conference on Computer Vision*, 2015, pp. 1026–1034.

[23] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," 2009.

[24] A. Das, S. Roy, U. Bhattacharya, and S. K. Parui, "Document image classification with intra-domain transfer learning and stacked generalization of deep convolutional neural networks," in *International Conference on Pattern Recognition*, 2018, pp. 3180–3185.

[25] A. W. Harley, A. Ufkes, and K. G. Derpanis, "Evaluation of deep convolutional nets for document image classification and retrieval," in *International Conference Document Analysis and Recognition*, 2015, pp. 991–995.