Embedded Textual Content for Document Image Classification with Convolutional Neural Networks

Lucia Noce, Ignazio Gallo, Alessandro Zamberletti and Alessandro Calefati
University of Insubria
Varese, Italy
lucia.noce@uninsubria.it

ABSTRACT

In this paper we introduce a novel document image classification method based on combined visual and textual information. The proposed algorithm's pipeline is inspired to the ones of other recent state-of-the-art methods which perform document image classification using Convolutional Neural Networks. The main addition of our work is the introduction of a preprocessing step embedding additional textual information into the processed document images. To do so we combine Optical Character Recognition and Natural Language Processing algorithms to extract and manipulate relevant text concepts from document images. Such textual information is then visually embedded within each document image to improve the classification results of a Convolutional Neural Network. Our experiments prove that the overall document classification accuracy of a Convolutional Neural Network trained using these text-augmented document images is considerably higher than the one achieved by a similar model trained solely on classic document images, especially when different classes of documents share similar visual characteristics.

Keywords

Document Image Classification; Convolutional Neural Network; Natural Language Processing

1. INTRODUCTION

Document image classification and retrieval is an important task in document processing as it is a key element in a wide range of contexts, such as: automated archiving of documents, Digital Library constructions and other general purpose document image analysis applications [4].

Nowadays a large number of documents are produced, processed, transferred and stored as digital images everyday: forms, letters, printed articles and advertisement are only few examples of them. While documents belonging to different macro-areas (*e.g.* financial, advertisement, *etc.*) tipically show substantially different visual layouts from one another and thus can be accurately classified just by comparing their visual characteristics; the same does not hold true for documents belonging to the same macro-area but different sub-areas (*e.g.* house ads, shop ads, *etc.*).

Given this situation, when the task is to perform a highly specific document classification among different document categories which are are both visually and semantically very similar, a combined content and visual analysis is mandatory to achieve satisfying fine-grained classification results. To this end, in this manuscript we propose a novel system to perform fine-grained document image classification exploiting both the content and the visual characteristics of the processed documents.

There are plenty of methods in literature that perform document classification relying exclusively on textual content extracted from the processed documents. While those algorithms can be effective for simple and well-made artificial documents, they have many limitations: (i) they ignore important visual document features (*e.g.* tables, images and diagrams) that may play an important role in the final document classification predictions, (ii) they are limited to printed documents due to Optical Character Recognition (OCR) limits, and (iii) they cannot be used to classify documents that do not contain any textual information or contain machine-unreadable text.

Accordingly, image analysis is complementary to content-based document classification for many classes of documents, and several techniques in literature successfully rely on structural and visual aspects to perform coarse-grained document classification [8].

For documents that present the same fixed structure, such as forms, template matching is adopted [25, 24]: for each class, a template is manually chosen and during the classification phase the input document is matched with one or more class-representative template. Various layout-based features are also adopted, and many works show their effectiveness for the document classification task [6, 1]. Following state-of-the-art results obtained in the Computer Vision research field, many recent works [8, 18] also perform document image classification using Deep Convolutional Neural Networks (CNN), obtaining outstanding results [8]. Starting from these state-of-the-art results and considering the previously described fine-grained document classification problem, we propose a novel method that combines textual and visual features using CNN.

More in details, our proposal consists of embedding content information extracted from text within document images, with the aim of adding elements which help the system in distinguish different classes that appear visually indistinguishable.

Our model was evaluated on two different dataset, and between different numbers and gender of document categories; the results from our experimental phase show that the proposed methodology achieves competitive results when compared to recent related

DocEng '16, September 12-16, 2016, Vienna, Austria

DOI: http://dx.doi.org/10.1145/2960811.2960814

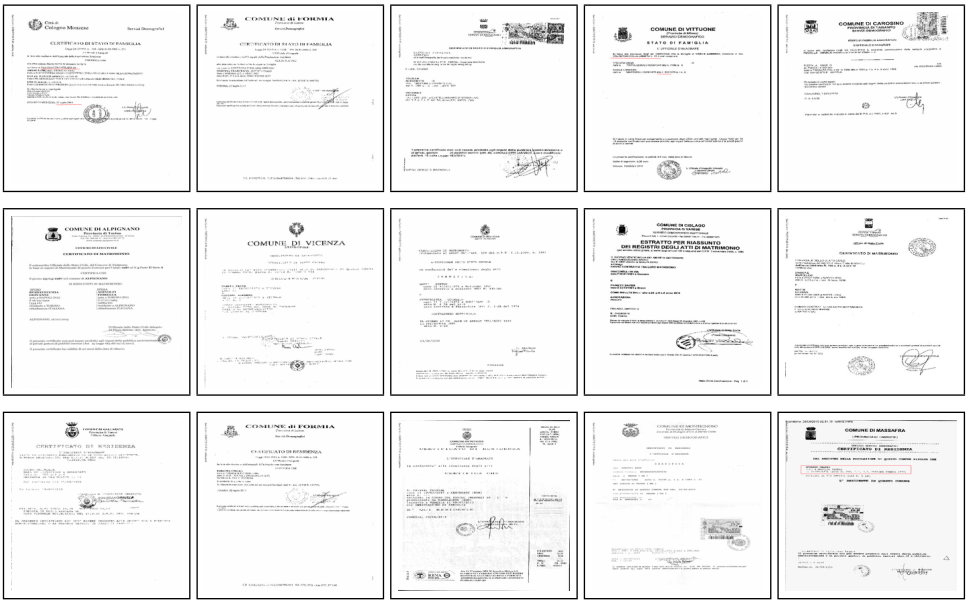


Figure 1: Documents belonging to three different representative classes are shown. In the first line examples of the class "Family Status" are shown, in the second line documents that belong to the class "Marriage Certificate" are reported, while in the third line documents were extracted from the "Residence Certificate" class. It's almost impossible to distinguish between the 3 classes only relying on the visual style of the documents.

works, and is able to effectively perform fine-grained document image classification.

2. RELATED WORKS

Existing approaches for document image classification and retrieval differ from each other based both on the type of extracted information (textual or visual) and/or the type of image analysis that is performed over the processed documents (global or local). Different supervised and unsupervised models have been proposed in literature throughout the last decade [4]: Random Forest based [10], Decision Tree and Self-organizing Map based [27], K-Nearest Neighbor based [3], Hidden Markov Model based [6], and Graph matching based [21] to name a few.

Features extracted from document images can either be visual, textual, or a combination of those two. The percentage of text and non-text elements in a content region of the image, font sizes, column structures, document structure, bag-of-words, and statistics of features are only 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 [26, 10, 4].

In literature, visual-based local document image analysis was investigated and adopted for document images classification [9, 25]. Region-based algorithms reach interesting results when applied to structured model, such as letters or forms. Classifying a document based on its whole visual content is also possible [27]. However, all of these cited visual feature based approaches have limitations, such as the manual definition of document templates or specific geometric configurations of fixed features related to different document layouts.

Content-based document classification has also been extensively studied in literature. Content-based analysis of documents is typically performed relying on text extracted using OCR methods, although text allows retrieving information about document content, visual layout plays an equal important role and it's used to detect some image region where applies OCR in order to have a more accurate extraction of content elements [2]. Nonetheless, OCR is prone to errors and is not always applicable to all kind of documents *e.g.* handwriting text is still difficult to read and those document images must have high resolution.

The recent success of Convolutional Neural Networks (CNN) in Computer Vision research areas [22, 32, 29] inspired novel applications of those algorithms to other domains such as document image analysis, text categorization and text understanding. A recent work by Kang *et al.* [18] shows that CNN are able to learn the entire supervised document image classification process, from feature extraction to final classification. Authors propose the use of a CNN for document image classification: a CNN is trained to recognize the class of given subsampled and pixel value normalized document images. Authors test their model on several challenging datasets, showing that such approach outperforms all previously explored methodologies.

Following the same intuition, Harley *et al.* [8] achieve outstanding results, setting new state-of-the-art results for image document image classification and retrieval.

An extensive evaluation of their CNN model is reported, determining that features extracted from deep CNN exceed the performance of all alternative visual and textual features both on document image classification and retrieval by a large margin. They also investigate about transfer learning, asserting that features learnt using CNN trained on object recognition are also effective in describing documents. Moreover, authors present several experiments varying between a single holistic CNN and ensembles of region-based CNN, exploring different initialization strategies. The performances of the evaluated models are evaluated on 2 different subsets of the IIT CDIP Test Collection [20], the smallest one [11] coincide with the one used by Kang *et al.* [18], while the largest one is composed by a significantly larger set of documents [8]. Evalu-

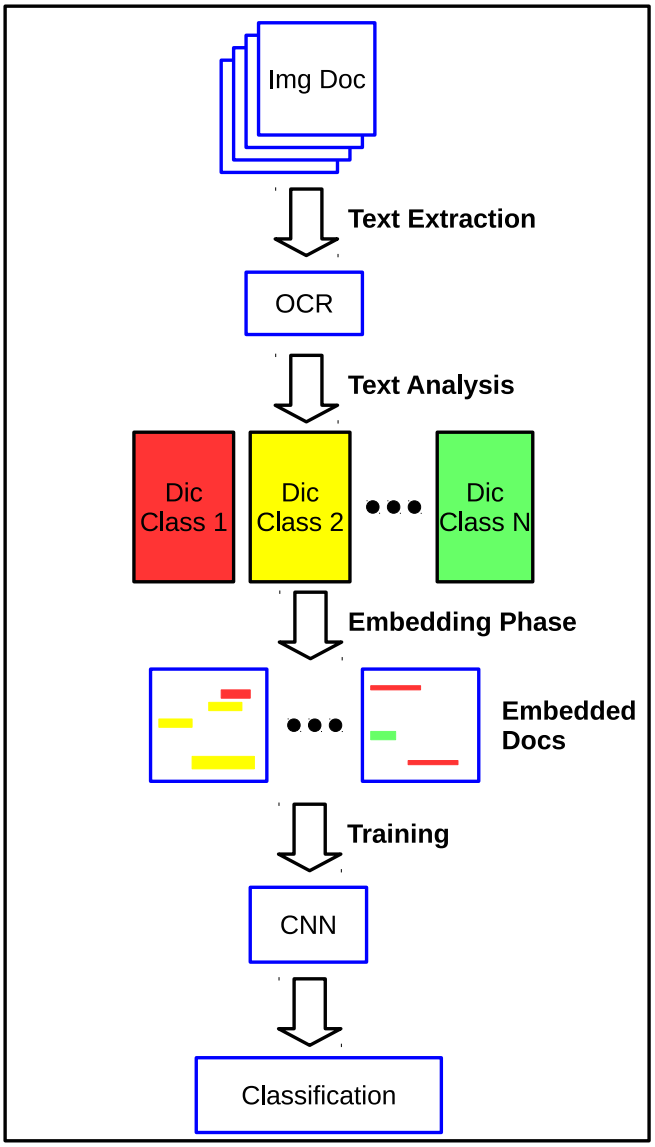


Figure 2: The building phase algorithm's pipeline.

ation against all the different CNN configuration and diverse bag-of-words approaches are reported.

CNN are also employed in text categorization and text understanding fields. The text obtained by applying OCR to a document image can be viewed as a document itself. In this manner, document image classification can be reconducted to a sentence-level classification task.

Several interesting works that use CNN for Natural language processing have been recently published. Kim *et al.* [16] propose a simple CNN with one Convolutional Layer and prove that it performs well when applied to a wide range of difficult text classification tasks such as sentiment analysis and question classification. CNN are not the only Deep Model that proved to be effective for document classification tasks; Zhang *et al.* [31] use other Deep Learning algorithms for several text understanding tasks using Temporal Convolutional Networks [19] (ConvNets), working with both English and Chinese languages. An other recent work by Johnson *et al.* [13] studies CNN for text categorization exploiting word order to lead to more accurate predictions.

Our work exploits CNN in the same manner as in the approaches proposed by Harley *et al.* [8] and Kang *et al.* [18], our aim is to combine both content-based and image analysis, and to do so we embed content information extracted from text into subsampled document images and feed those visually enriched documents to a properly trained CNN model. A comparison between most of the previously cited approaches is provided and demonstrate that our model can easily reach comparable results. We also compare our results applying CNN only to text extracted from document images following the same approach of Kim *et al.* [16]. Results show that even though content-based approach leads to accurate results, our proposal that uses both images and text information performs even better. This demonstrates that the combination of the two different genre of features (visual and content) allows Machine Learning models to reach higher accuracy values in difficult document image classification tasks especially in the case of high visual similarity between different classes of documents.

3. PROPOSED METHOD

The main idea behind this work comes out after considering that in document image classification, intra-class similarity represents an issue that can be solved adding textual information extracted from the processed document images.

Figure 1 shows some representative examples of intra-class similarity, if in our method, we just focused on visual features, it would have been impossible to distinguish between the two classes of documents shown. However, by underlining significant textual information, fine-grained document classification become easier.

With the currently available computational resources, the adoption of CNN only allows the use of small sized document images. Although the original document layouts are distinguishable even sub-sampled document images, text becomes unreadable. Our challenge consists of adding content information in a visual manner, to let the CNN model to exploit such information when performing the classification task to reach higher classification accuracies for documents having high intra-class similarity.

The pipeline of the building phase of proposed approach is shown in Figure 2, its three main phases can be summarized as follows:

- **Text extraction and analysis**: OCR is employed to extract textual information from original sized document images. Texts are analyzed and for each class a dictionary is built.
- **Embedding phase**: exploiting word position coordinates and dictionaries, relevant words are emphasized within each sub-sampled document image.
- **Training phase**: a CNN is trained using sub-sampled document images from the previous phase.

3.1 Text extraction and analysis

Textual information is extracted from each document image through OCR. We employ Tesseract OCR [28], a widely used open-source OCR engine[1].

Optical Character Recognition is a difficult task for noisy or low resolution documents [14], and thus to discard reading errors we preprocess all the automatically extracted text using Natural Language dictionaries (more details about the used Natural Languages will be given in Section 4) and stop-word lists.

[1]https://github.com/tesseract-ocr

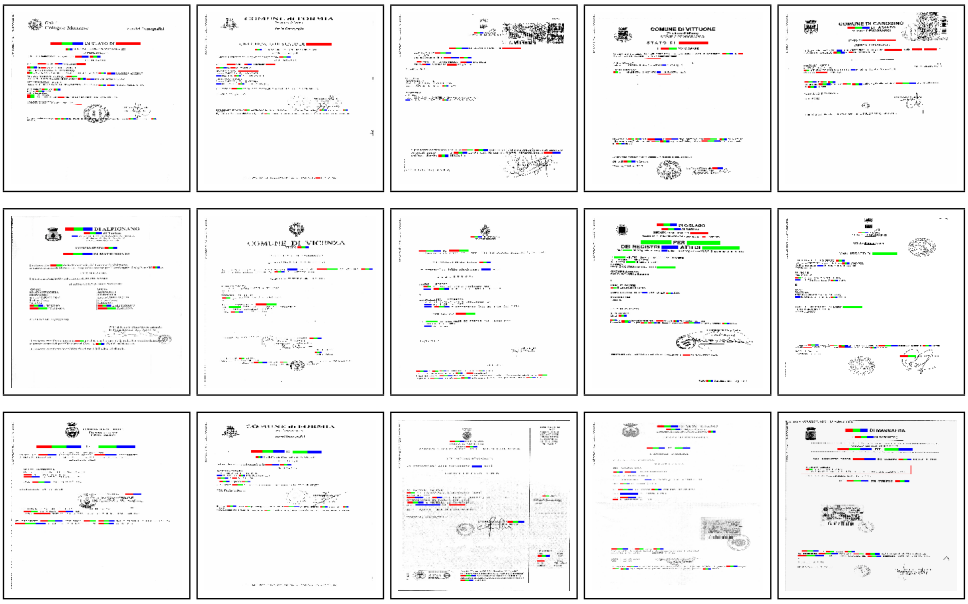


Figure 3: Key-words of three classes are underlined within images. It is easier to distinguish between the three classes "Family Status", "Marriage Certificate" and "Residence Certificate". The colors red, green and blue emphasize the content information in images, making it available for the training and classification phases.

To emphasize class relevant textual content within each document image, for each class, a dictionary containing representative words is generated. This is done by collecting all the words extracted by the OCR engine, for all the images belonging to a specific class. To build the final dictionary, we adopt the weighting formula of Peñas *et al.* [23].

The relevance formula, associates a weight to each word comparing it to other classes' words, the more the higher the value the higher is the relevance of the word for the specific class. In detail, the relevance of a term t_i is computed as follows:

$$Relevance(t_i, sc, gc) = 1 - \frac{1}{\log_2\left(2 + \frac{F_{t_i,sc} \cdot D_{t_i,sc}}{F_{t_i,gt}}\right)} \qquad (1)$$

where: (i) *sc* is the *specific corpus*, it corresponds to the subset of words, extracted from the starting set of words extracted from images of the specified class (ii) *gc* is the *generic corpus*, it is composed by the whole set of words, extracted from all the classes , (iii) $F_{t_i,sc}$ is the relative frequency of the term t_i in the specific corpus *sc*, (iv) $F_{t,gc}$ is the relative frequency of the same term t_i in the generic corpus *gc* and (v) $D_{t,sc}$ is the relative number of documents of *sc* in which the term t_i appears.

Once we have the relevance value associated to each word of each dictionary, we prune the set of word at the fixed threshold $r = 0.8$ where r has been empirically determined.

Following all the previously described steps, we obtain a set of dictionaries that represent classes' key-words that are used to underline text within images in the next step.

3.2 Embedding phase

Starting from the dictionaries of relevant words per classes, the aim of this phase is to embed textual information obtained from OCR within document images. We perform this to let relevant key-words information become recognizable even at low resolutions where the text is unreadable.

We create a specific visual color feature for each class key-word contained in the processed image and in at least one of the dictionaries built in Sec. 3.1. The added visual feature consists of a rectangle of the class color drawn across each class key-word found in the document image.

More in details, given an image, Tesseract OCR is performed. The OCR engine output is composed both of the sequence of recognized words and their positions within the image. Once the words are extracted, for each word the system checks whether it belongs to one or more of class key-word dictionaries.

If the word belongs to only one dictionary, a rectangle of the associated class color is drawn across it using the obtained position coordinates, otherwise if it belongs to more than one dictionary the rectangle is divided by the number of corresponding dictionaries and each part is colored using the associated class colors.

In Figure 3 the same documents shown in Figure 1 are shown after the embedding phase. Rectangles of respective classes' colors are drawn; it can be easily noted that, for documents that belong to the same class, the associated class color is the mostly used: in the first line the red color is the most utilized, in the second is the green, while in the third the majority of the key-words' rectangles are blue. Experiments reported in Section 4 show the effectiveness of the embedding phase for these specific three classes of documents.

During CNN training, document images are sub-sampled to fixed dimension therefore text becomes unreadable; however, the marked key-word rectangles remain visible and allow the model to infer textual content. Not only classes information are added but key-words' positions are underlined, giving the model extra characteristics that are exploited during the classification phase.

3.3 Training phase

A deep Convolutional Neural Network is employed as classification model. Our proposal consists of using the images from the previous steps, where textual content information are transformed into visual features and stored in document images, to train the network.

A common practice with CNN is to exploit transfer learning [30].

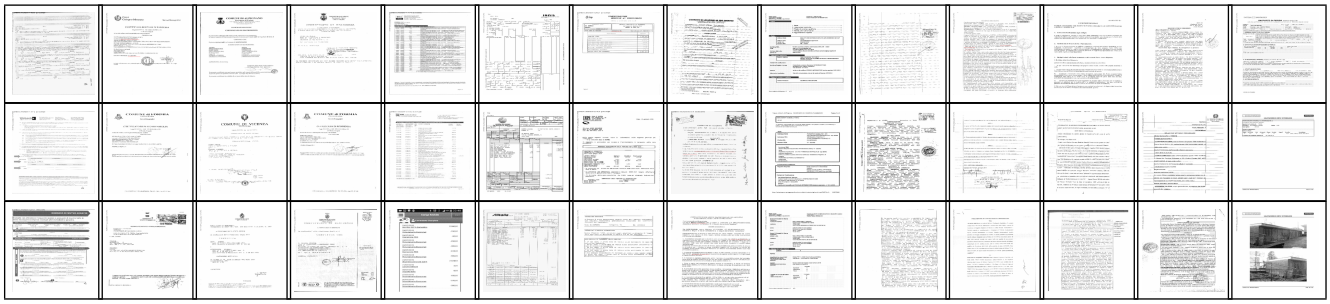


Figure 4: The Loan dataset: for every column, three representative documents of a specif class are displayed. From left to right classes are reported in the following order: "Loan Request", "Family Status", "Marriage Certificate", "Residence Certificate", "Account Balance", "Payroll", "Pension Payslip", "Lease", "Company Registration", "Previous Contract", "Preliminary Purchase", "Loan Contract" ,"Preliminary Report" and "Expertise". It can be noted that documents belonging to different classes present similar visual style.

This technique consist of pre-training a network on a large dataset, and then exploit it either as a fixed feature extractor or as a fine-tuning for the adopted CNN. In the first scenario, given a CNN a training phase on a different dataset is performed, after that the last fully-connected layer is removed and the remaining Convolutional Network is treated as a fixed feature extractor for the new dataset. On the other hand, the second strategy consists of fine-tune the weights of the pre-trained network by continuing the back-propagation.

A popular pre-training dataset is ImageNet. ImageNet dataset [5] is composed of over 15 million labeled high-resolution images in over 22000 categories, a subset of 1.2 million of images divided in 1000 categories is used in ILSVRC ImageNet challenges as training set, networks trained with such a training set are often used for transfer learning methodology.

Supported by the state-of-the-art results obtained by Harley *et al.* [8], we also implement transfer learning using ImageNet dataset and the CNN model of Krizhevsky *et al.* [17]. Implementation details are given is Section 4.2. Multiple experiments demonstrating the effectiveness of our method are reported in the next Section of this manuscript.

4. EXPERIMENTS

Several experiments were conducted in order to evaluate our proposal. As discussed in previous Sections, we aim to add textual content information to document images, to allow the CNN model to better distinguish between classes that have hight intra-class visual similarity.

We compare our results against state-of-the-art outcomes in the document image classification field, and provide a comparison between a CNN that performs classification using the proposed textual content embedded document images and a CNN that uses exclusively textual information, showing that our methodology is significantly more effective.

4.1 Datasets

We test our approach on two datasets, and to better emphasize the effectiveness of our proposal in solving intra-class similarity issue, we use some representative subsets of the same document image collections.

The first dataset, the so-called *Loan* dataset, has been provided by an Italian loan comparison website company. Through their platforms, this company provides a service which let customers quickly compare the best rates and terms of available loans. To supply the best obtainable loan, website owners need to collect all the necessary documents in a digital format.

We collect a set of 14 different classes that present similar visual styles, this means that a document belonging to a class is often indistinguishable using just visual style features. Documents are used to provide loan offers to customers. A total of 16250 document images are collected and divided into training, evaluation and testing sets following similar proportions as in the ImageNet dataset: 80% training, 10% evaluation and 10% testing. Examples of image documents extracted from the different classes are reported in Figure 4.

Among all the 14 collected classes we select 2 subsets of 3 classes each that have very similar visual layouts and because of that are difficult to classify correctly. Visually similar subset are used to better underline the effectiveness of our proposal.

The first subset is composed by the following classes: "Family Status", "Marriage Certificate" and "Residence Certificate", previously introduced in Figure 1 and 3; we name it *Certificates subset*. The second subset is called *Contracts subset* and it contains the following document classes: "Preliminary Purchase", "Loan Contract" and "Preliminary Report".

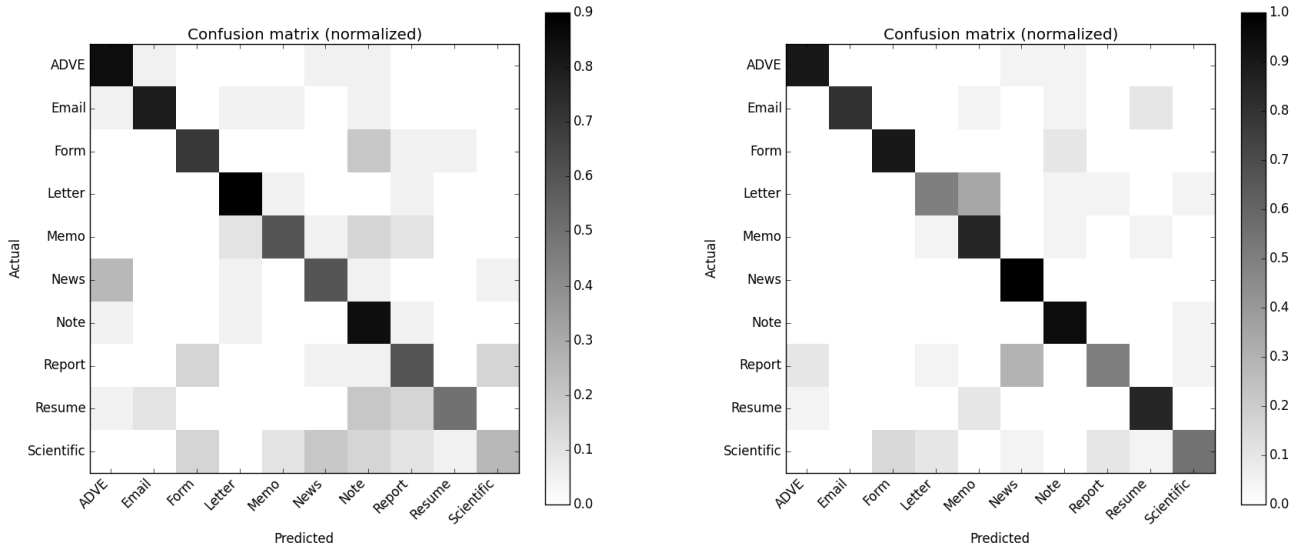
In order to compare our approach to existing approaches, we test our method on the same dataset used by Harley *et al.* [8] and Kang *et al.* [18], we refer to it as *Tobacco* dataset. The *Tobacco* dataset is composed of 3482 images divided in the following 10 classes: Advertisement, Email, Form, Letter, Memo, News, Note, Report, Resume, Scientific.

The split of Tobacco dataset is the same used in related works [8, 18, 20]: 800 images are used for training, 200 for validations, and the remainder for testing.

4.2 Implementation details

The CNN is implemented using Caffe[12], and is based on the work of Krizhevsky *et al.* [17]. The network is composed of 5 Convolutional Layers, some of which are followed by Max-pooling Layers, three Fully-connected Layers with a final Softmax, all the details are reported in the reference paper [17].

The full architecture can be written as 227×227 – $11 \times 11 \times 96$ – $5 \times 5 \times 256$ – $3 \times 3 \times 384$ – $3 \times 3 \times 384$ – $3 \times 3 \times 256$ – 4096 – 4096 – N. Where N is the number of categories and varying in respect with the utilized dataset. The input images size is 227×227, we down-sampled all the images to a fixed resolution of 256×256, in order to have a constant input dimensionality. The Caffe implementation of the adopted CNN randomly crops images from



(a) Visual Features

(b) Textual and Visual Features

Figure 5: Confusion matrices reporting the results achieved on the Tobacco dataset. Results have been calculated testing the model trained using original images (a) and using images that contained textual information (b).

Table 1: Results achieved by the proposed method are reported, the CNN was trained using original images and with elaborated images. More accurate results are obtained training the network with images in where textual information are embedded.

Dataset	Overall Accuracy
Tobacco V	74.2%
Tobacco V & T	79.8%
Loan V	74.73%
Loan V & T	87.85%
Loan Certificates subset V	60%
Loan Certificates subset V & T	90%
Loan Contracts subset V	62.31%
Loan Contracts subset V & T	88.33%

256×256 to 227×227, this technique is usually employed because it helps data augmentation and reduces overfitting.

4.3 Results

Experiments are performed to compare different results achieved while training the CNN model with original images and images containing textual information created through the embedding phase (Section 3.2).

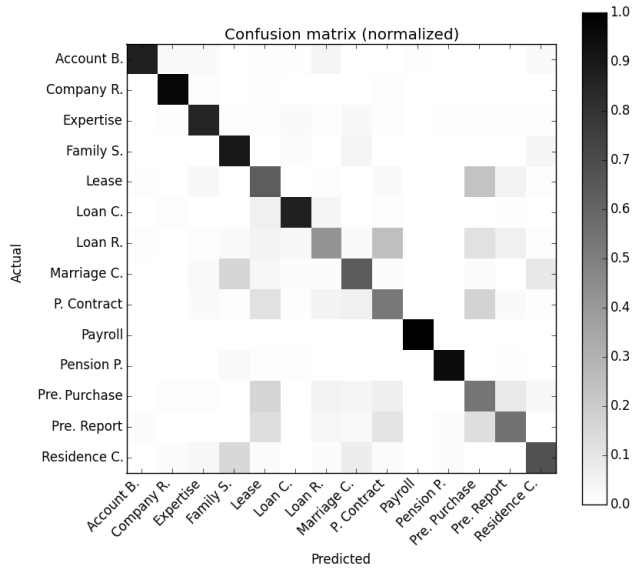
Overall accuracies achieved by the proposed model are reported in Table 1. The adopted CNN was trained using the two different types of images. Results demonstrate that when the CNN is trained on the embedded images it reaches higher accuracies, passing from 74.2% to 79.8% on the Tobacco dataset and reaching 87.85% from 74.73% on the Loan dataset. Figure 5 shows the two confusion matrices computed for the Tobacco dataset respectively before and after the embedding phase; matrices underline that information deriving by text is helpful for the classification task and show the achieved improvement.

Table 2: The same experiments performed exploiting visual features are performed using only textual features. Results shows that text is a relevant feature for the selected classes and can not be ignored to better perform the classification task, but at the same time a combinations of textual and visual feature is more effective.

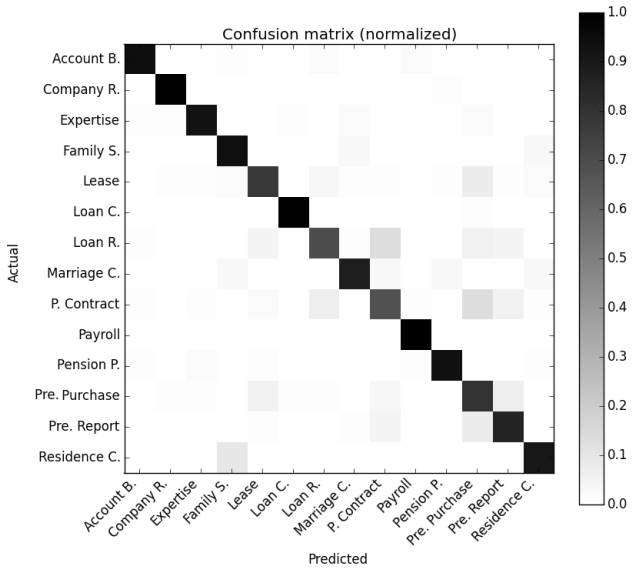
Dataset	Text CNN	Proposed
Loan	69.89%	87.85%
Loan Certificates subset	87.15%	90%
Loan Contracts subset	86.31%	88.33%

Table 3: Comparison against models that use only visual features or only textual features are reported, our proposal overcomes or reaches almost the same results achieved by other methods.

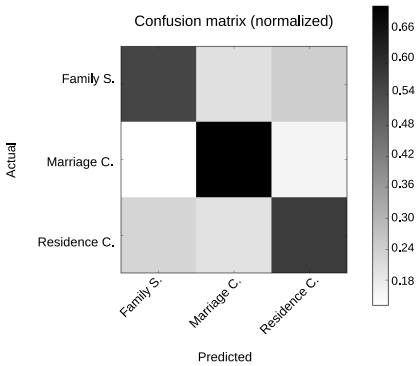
Model	Overall Accuracy
Proposed V & T	79.8%
text-CNN Kim [16]	68.92 %
CNN Harley *et al.* [8]	79.9 %
CNN Kang *et al.* [18]	65.35%



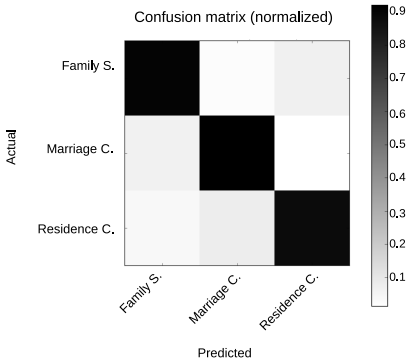
(a) LOAN - Visual Features



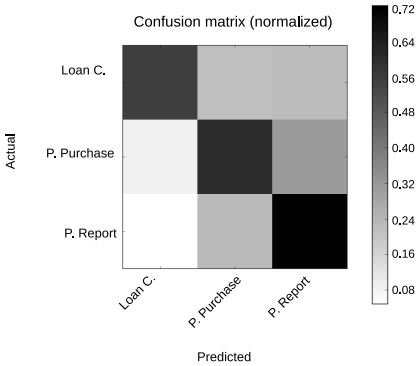
(b) LOAN - Textual & Visual Features



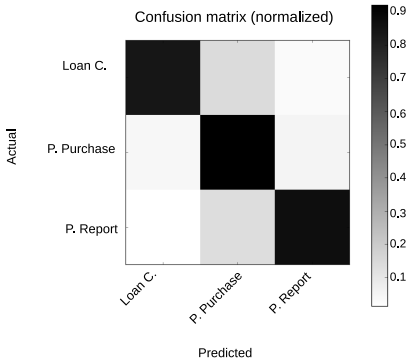
(c) CERTIFICATES - Visual Features



(d) CERTIFICATES - Textual & Visual Features



(e) CONTRACTS - Visual Features



(f) CONTRACTS - Textual & Visual Features

Figure 6: Confusion matrices reporting results achieved on the whole Loan dataset (a), (b) and on the two subsets Certificates (c), (d) and Contracts (e), (f) are displayed. Matrices on the left side report the outcomes of the test phase, training the model using original images, while on the right side, the model was trained using images that contains underlined textual information. Matrices summary the improvement achieved by using the elaborated images in the training phase.

Experiments conducted on the two subsets of the Loan dataset show a high grow in terms of accuracy, the values increase by 30% passing from original images to textual embedded document images for the Certificates subset, and by 26% for the Contracts subsets.

Confusion matrices of the testing phase for both the Loan and the two subsets are provided. Figure 6 displays the confusion matrices related to the Loan dataset and its two subsets, all the three matrices reflect the overall accuracy values obtained, demonstrating the effectiveness of the propose methodology.

Moreover Table 3, reports a comparison carried on the Tobacco dataset among our proposal and related works that deals with only visual features[8, 18] or only textual features [16], which will be better analyzed in Section 4.4.

Although the proposed method does not overcome state-of-the-art results, it reaches comparable outcomes and demonstrates that combining visual and textual feature is an interesting approach to follow.

4.4 CNN applied to text

In this work, we aim to classify images adding textual information to them. Our goal is to demonstrate that a combination of textual and visual features is necessary for better understanding the document image content. To this end, we evaluate the classification of images using just the extracted text.

CNN proved effective in different Natural Language Processing tasks [15, 7]. A recent work proposed by Yoon Kim [16] implements a simple CNN, with one Convolutional Layer and achieves good classification performance across a range of text classification tasks.

We use the model proposed by Kim *et al.* [16] to classify text extracted from document images. For each document, text is divided into sentences, each sentence represents a document that has to be classified into one of the available classes. Results are shown in Table 2 and demonstrate that although the text-CNN reaches interesting results achieving 70% of accuracy on the whole Loan dataset, the exploitation of both visual and textual feature is more effective.

5. CONCLUSIONS

A new method that exploits both textual and visual features for document image classification has been proposed. By adopting Convolutional Neural Networks, we demonstrate that embedding textual information into document images leads to more accurate results for the document image classification task. Our method is able to take advantage of the extra textual key-word information provided by colored rectangles to reach more satisfying document image classification accuracies, especially for document classes having similar visual styles.

Future extensions of our work may rely on testing different embedding methods that permit to apply this approach on classification tasks that involve a consistent number of classes.

6. ACKNOWLEDGMENTS

We gratefully acknowledge the support of NVIDIA Corporation with the donation of the GeForce GTX 980 GPU used for this research.

7. REFERENCES

[1] E. Appiani, F. Cesarini, A. M. Colla, M. Diligenti, M. Gori, S. Marinai, and G. Soda. Automatic document classification and indexing in high-volume applications. *International Journal on Document Analysis and Recognition (IJDAR)*, 2001.

[2] S. Argamon, O. Frieder, D. A. Grossman, and D. D. Lewis. Content-based document image retrieval in complex document collections. In *Document Recognition and Retrieval (DRR)*, 2007.

[3] S. Baldi, S. Marinai, and G. Soda. Using tree-grammars for training set expansion in page classification. In *International Conference on Document Analysis and Recognition (ICDAR)*, 2003.

[4] N. Chen and D. Blostein. A survey of document image classification: Problem statement, classifier architecture and performance evaluation. *International Journal on Document Analysis and Recognition (IJDAR)*, 2007.

[5] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. ImageNet: A Large-Scale Hierarchical Image Database. In *The IEEE Conferenbasedce on Computer Vision and Pattern Recognition (CVPR)*, 2009.

[6] M. Diligenti, P. Frasconi, and M. Gori. Hidden tree markov models for document image classification. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 2003.

[7] C. dos Santos and M. Gatti. Deep convolutional neural networks for sentiment analysis of short texts. In *International Conference on Computational Linguistics (COLING)*, 2014.

[8] A. W. Harley, A. Ufkes, and K. G. Derpanis. Evaluation of deep convolutional nets for document image classification and retrieval. In *International Conference on Document Analysis and Recognition (ICDAR)*, 2015.

[9] J. Hu, R. Kashi, and G. Wilfong. Comparison and Classification of Documents Based on Layout Similarity. *Information Retrieval*, 2000.

[10] Jayant Kumar and David Doermann. Unsupervised Classification of Structurally Similar Document Images. In *Intl. Conf. on Document Analysis and Recognition (ICDAR 13)*, 2013.

[11] Jayant Kumar, Peng Ye, and David Doermann. Structural Similarity for Document Image Classification and Retrieval. *Pattern Recognition Letters*, 2013.

[12] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell. Caffe: Convolutional architecture for fast feature embedding. In *ACM International Conference on Multimedia (ACMMM)*, 2014.

[13] R. Johnson and T. Zhang. Effective use of word order for text categorization with convolutional neural networks. *Computing Research Repository (CoRR)*, 2014.

[14] A. Kae, G. B. Huang, C. Doersch, and E. G. Learned-Miller. Improving state-of-the-art OCR through high-precision document-specific modeling. In *The IEEE Conferenbasedce on Computer Vision and Pattern Recognition (CVPR)*, 2010.

[15] N. Kalchbrenner, E. Grefenstette, and P. Blunsom. A convolutional neural network for modelling sentences. *Computing Research Repository (CoRR)*, 2014.

[16] Y. Kim. Convolutional neural networks for sentence classification. *Computing Research Repository (CoRR)*, 2014.

[17] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In

Advances in Neural Information Processing Systems (NIPS), 2012.

[18] Le Kang, Jayant Kumar, Peng Ye, Yi Li, and David Doermann. Convolutional Neural Networks for Document Image Classification. In *International Conference on Pattern Recognition (ICPR)*, 2014.

[19] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. In *IEEE*, 1998.

[20] D. Lewis, G. Agam, S. Argamon, O. Frieder, D. Grossman, and J. Heard. Building a test collection for complex document information processing. In *ACM Conference on Research and Development in Information Retrieval (SIGIR)*, 2006.

[21] J. Liang, D. S. Doermann, M. Y. Ma, and J. K. Guo. Page classification through logical labelling. In *International Conference on Pattern Recognition (ICPR)*, 2002.

[22] M. Liang and X. Hu. Recurrent convolutional neural network for object recognition. In *The IEEE Conferenbasedce on Computer Vision and Pattern Recognition (CVPR)*, 2015.

[23] A. Penas, F. Verdejo, and J. Gonzalo. Corpus-based terminology extraction applied to information access. In *Corpus Linguistics*, 2001.

[24] H. Peng, F. Long, Z. Chi, and W.-C. Siu. Document image template matching based on component block list. *Pattern Recognition Letters*, 2001.

[25] P. Sarkar. Learning image anchor templates for document classification and data extraction. In *International Conference on Pattern Recognition (ICPR)*, 2010.

[26] C. Shin and D. S. Doermann. Document image retrieval based on layout structural similarity. In *International Conference on Image Processing, Computer Vision, and Pattern Recognition (IPCV)*, 2006.

[27] C. Shin, D. S. Doermann, and A. Rosenfeld. Classification of document pages using structure-based features. *International Journal on Document Analysis and Recognition (IJDAR)*, 2001.

[28] R. Smith. An overview of the tesseract OCR engine. In *International Conference on Document Analysis and Recognition (ICDAR)*, 2007.

[29] R. Wu, B. Wang, W. Wang, and Y. Yu. Harvesting discriminative meta objects with deep CNN features for scene classification. *Computing Research Repository (CoRR)*, 2015.

[30] J. Yosinski, J. Clune, Y. Bengio, and H. Lipson. How transferable are features in deep neural networks? *Computing Research Repository (CoRR)*, 2014.

[31] X. Zhang and Y. LeCun. Text understanding from scratch. *Computing Research Repository (CoRR)*, 2015.

[32] Y. Zhang, K. Sohn, R. Villegas, G. Pan, and H. Lee. Improving object detection with deep convolutional networks via bayesian optimization and structured prediction. *Computing Research Repository (CoRR)*, 2015.