The usage of U-Net for pre-processing document images

Diandra A. A. Kubo, Tiago S. de Nazare, Priscila L. R. Aguirre, Bruno D. Oliveira, Felipe S. L. G. Duarte
Data Science Team Itaú Unibanco
{diandra.kubo, tiago.nazare, priscila.aguirre, bruno.domingues-oliveira, felipe.duarte}@itau-unibanco.com.br

***Abstract*—When processing documents in real-world scenarios, it is common to deal with artifacts that may hamper document analysis, such as stamps, noise and strange backgrounds. Aiming to mitigate these problems, we propose the use of U-Net, a very successful biomedical image segmentation network, for handwritten and machine text segmentation. In order to do so, we trained a model for each type of text. One of the main advantages presented is that the models are trained on artificial data, avoiding the wearisome task of data labeling. For the machine text segmentation model, we test its impacts on both word and character recognition when combined with the Tesseract OCR model. For the handwritten segmentation model, we present qualitative results. Initial experiments indicate that both models are able to improve results in their respective applications.**

I. Introduction

Recent advances in machine vision techniques allowed the usage of models to solve real-world problems such as image classification [1], object detection [2], signature verification [3] and optical character recognition (OCR) [4]. Despite such progress, in most cases, the impacts that image quality can have on performance is overlooked, therefore models are still limited with regards to their applicability in practical situations where images obtained in unconstrained scenarios [5].

When a document is scanned by a sensor and converted to a digital image, some noise can be inherited in this process. The scanner may perform some kind of pre-processing technique (e.g. thresholding, quantization, filtering and compression) in order to save memory and computational effort, which can lower the image resolution, hence quality [6].

Furthermore, the variety of components that a document can contain also presents itself as a challenge. Some of these might include handwritten text, lines, machine-made text and stamps. As pointed out in [7], the applications derived from these documents analysis all rely on the correction of undesired artifacts and the enhancement of specific chosen features.

One of the main applications in document analysis is OCR, which consists of converting image text into machine readable text, that is string. For this type of problem there are many of-the-shelf solutions that perform relatively well (e.g. Tesseract) [4]. However, most of these solutions assume a well behaved scenario, which might not always be the case in real applications, and therefore may present an issue. Thus, pre- and post-processing are the steps to be tackled when considering performance boosting for such applications. The major issue when using OCR techniques is the high variability in page layout and design, due to different document production processes [6].

When dealing with documents images, another recurring application is detecting the number of signatures in a document, finding their relative position on the page and who they belong to. This step depends on the correct extraction of possible signatures [8], detecting its locations and cleaning any noise surrounding them. On real-word applications, this is a complex problem [9], due to the variety of background and artifacts a document may present, such as stamps that might overlap the signature to be recognized.

Motivated by those gaps, in this paper, we propose a new framework for pre-processing document images using U-Net in two scenarios: (i) segmentation of machine-made text to improve OCR performance and (ii) extraction of handwritten text to help in applications such as signature verification. Our initial results show that U-Net can greatly improve in both character and word recognition when combined with Tesseract for the OCR task. Also, we present some images that indicate that this method can also be helpful when segmenting signatures (and other handwritten texts).

II. Technical Background

The U-Net architecture [10], was originally proposed for biomedical images and was very successful when dealing with cell segmentation on a pixel level. A great advantage presented by this model is that is fully convolutional, and thus can operate in any image size and therefore, is very suitable for this work's scenario.

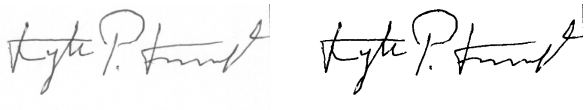
For computer vision applications, especially in segmentation tasks, finding reliably labeled data can be challenging, for it implies the need of a pixel-wise label in each image. Up until recently, such task is done manually by humans, which can be extremely time-consuming. This issue can be bypassed by generating synthetic images, as done in [11]. In this approach, ground truth is known from the start, and any variability can be added to the data, making it as complex as desired [12], which can be advantageous.

III. Framework methodology

Achieving a good performance using any machine learning method depends on the quantity and quality of the input

data. Usually, most algorithms require a great number of observations, that must also be carefully labeled [13]. Due to the lack of a large and labeled dataset needed for a proper training in the area, we created our own artificial dataset and train two U-Nets, one for retrieving handwritten text, and a second one that maintains machine written text.

To replicate the handwritten text typically found in documents, we used the CEDAR signature database [14], in which 55 participants were asked to sign their names 24 times. They also were asked to simulate another 3 signatures given, 8 times each, resulting in a database with 1320 images of genuine signatures, and 1320 forgeries. Yet, only forged signature images were used in our dataset, in order to increase variance. An Otsu threshold [15] was applied to the signatures in order to binarize and partially remove noise. An example can be seen in Figure 1.



(a) (b)

Fig. 1. Forged signature from the CEDAR database before (a) and after (b) otsu threshold.

Another artifact often found in documents are stamps. We reproduce this behavior using two images designed by Freepik from flaticon.com, that mimic real world stamps, as seen in Figure 2.

(a) (b)

Fig. 2. Stamps used on data generation.

Moreover, real-life documents usually contain tables, lines and bounding boxes, which should also be differentiated from text. With this in mind, we recreate such conditions in our dataset, adding random lines as well as complete grids and bounding boxes surrounding signatures.

We also apply other label-preserving transformations as a way to enlarge and build up complexity and variability of our training set [16]. Stamps can vary greatly due to the amount of ink used in the process. In order to replicate such behavior, the stamps are randomly resized and rotated, and can either: 1) be dilated, 2) eroded or 3) have a "salt" filter applied on them.

For the machine text used, we choose in a random way some font available in the operating system, and vary the font size between 5pt and 50pt.

Afterwards, we have a noisy image and a label image, where only the target type of text is shown, with the background set to 0. Examples of our generated dataset can be seen in Figure 3.

(a) (b)

(c) (d)

Fig. 3. Examples of generated images for machine text and handwritten text segmentation, where a) and c) are input images and b) and d) are their labels.

IV. Experiments

A. OCR Improvement

To test the machine text segmentation, we used the Tesseract OCR engine [17]. For such reason, as previously stated, the OCR model is treated as black box because retraining the model also requires a great deal of labeled data. Hence, an easier alternative to obtain improvements in text-detection is to apply pre-processing methods prior to OCR.

In order to evaluate OCR's quality improvment, we resorted to an open-source tool, named *ocveralUAtion*[1] [18], which computes the quotient between the number of mistakes and the text length. It is important to mention that the ground truth text is needed to compute the number of mistakes.

The tool provides per-character accuracy rates and word recognition rate. The former, referred to as *character error rate* (CER) is computed by

$$\text{CER} = \frac{i + s + d}{n}, \tag{1}$$

where n is the total number of characters, i is the minimal number of character insertions, s is the number of substitutions and d is the number of deletions d required to transform the reference text into the OCR output.

[1]Available at https://github.com/impactcentre/ocrevalUAtion

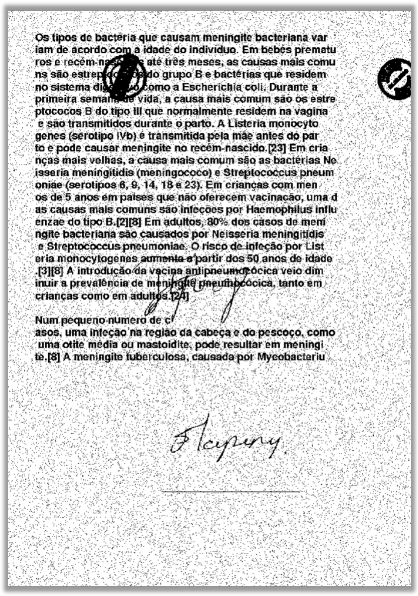


Fig. 4. Example of an image generated to test OCR performance.

Likewise, the word recognition rate is referred to as *word error rate* (WER) and is computed by

$$\text{WER} = \frac{i_w + s_w + d_w}{n_w}, \tag{2}$$

where n_w is the number of words in the reference text, s_w is the number of words substituted, d_w is the number of words deleted and i_w the number of words inserted required to transform the output text into the target, chosen such as to ensure that the sum $i_w + s_w + d_w$.

To test the trained network, we also created another artificial dataset. As we aimed to test OCR improvement in document analysis, we generated images that resemble real world documents. In order to evaluate the word recognition rate in addition to the character recognition rate, we needed text containing semantically correct sentences. For this reason, we resorted to collecting text from 14 articles on wikipedia [2]. This way, each portion of text used for all images generated was initially know and stored.

The text was placed respecting the idea of a border. Following, we added random noise to the image on account of replicating image complexity from real world scenarios. Hence, we added a signature on the bottom with a line near it, another signature on top of the document, and chose up to 4 stamps to be placed randomly on the document. Finally, a salt and pepper noise was applied. As a result, we had 1000 noisy images where Tesseract OCR was applied before and after our model. An example of this dataset can be seen in Figure 4.

B. Handwritten Text Recognition

Conventionally, research focus on signature recognition and validation, already assuming an optimal segmentation of the signature [9]. As explained, that may not be the case on real life settings, and therefore a pre-processing of the signature region is need. However, we currently are at the step of testing our model on segmentation alone, not evaluating yet its impact on signature recognition.

[2]https://wikipedia.org/

V. Results

The results from the OCR evaluation can be seen in Figures 5 and 6. These figures show the frequency distribution of the error rate's complement, described on Section IV-A. The results from the OCR applied on the original images are shown in orange, and the results from after the application of our model are shown in green.

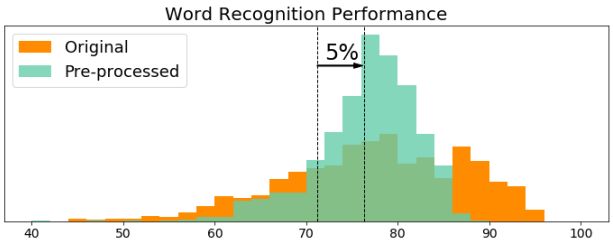


Fig. 5. Word recognition performance improvement after U-Net prediction

The improvement achieved in the word recognition are depicted in Figure 5. We can see a 5% improvement on the frequency average, and the distribution from the original images is more evenly distributed.

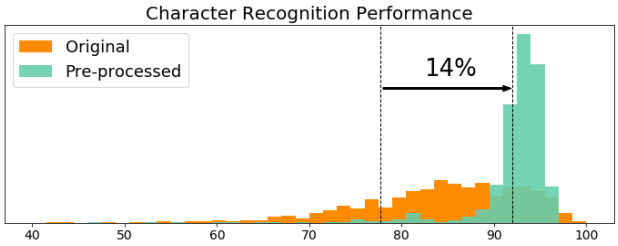


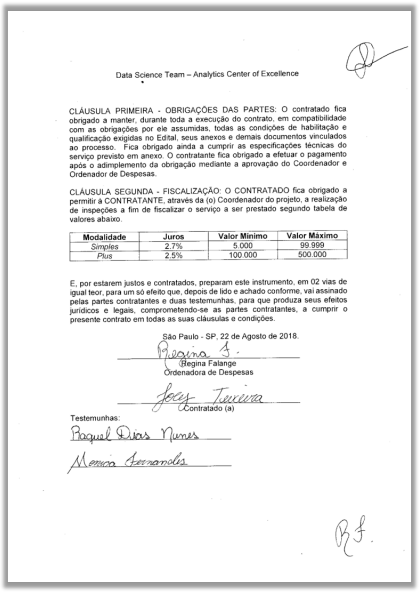
Fig. 6. Character recognition performance improvement after U-Net prediction

Figure 6 shows the results for the character recognition. A 14% upgrade on the frequency average was seen after the pre-processing by our model. The distribution from the processed images is unbalanced, being concentrated on higher scores.

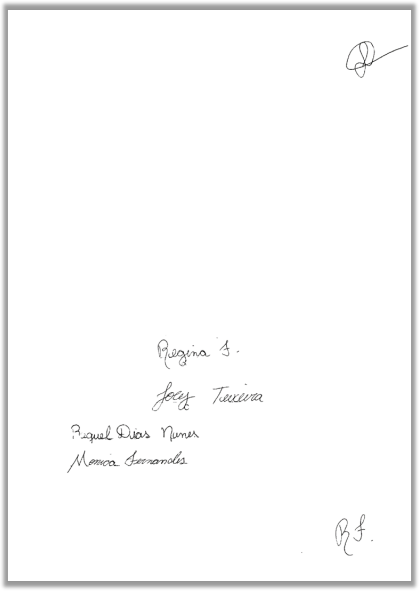
A qualitative result from our handwriting segmentation method can be seen in Figure 7. It can be seen that our method shows a good initial result removing machine text and tables which are typically found in documents being analyzed.

VI. Conclusion

In this paper, we proposed the usage of U-Net as pre-processing for two document analysis applications: OCR and signature verification. One of the main advantages of our approach is the generation of artificial training data, because we tried to replicate more realistic scenarios by adding noise and other artifacts (such as stamps and lines) to the image.



(a)



(b)

Fig. 7. Example of our model applied on a document, with a) being the original image and b) the image processed by our model.

Next, we conducted experiments to measure the impacts of such models in both applications.

According to our initial results, our pre-processing technique is very suitable for real scenarios with heavier image degradation. To evaluate our machine text segmentation, we measured performance improvements of applying this technique prior to using Tesseract. Our experiments show a 14% improvement in character detection and 5% improvement in word detection. With regards to handwritten text segmentation, we have shown some qualitative results that indicate that U-Net is also very adequate for this task.

As future research, we intend on tackling the following points:

- Further investigate the impacts of the machine text model on OCR detection by analyzing its results in more realistic data;
- Measure the impacts of the handwritten text segmentation model in signature verification.

ACKNOWLEDGMENT

The authors would like to thank Itaú-Unibanco for supporting this work.

REFERENCES

[1] G. Huang, Z. Liu, L. v. d. Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[2] J. Redmon and A. Farhadi, "YOLO9000: better, faster, stronger," in *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017*, 2017.

[3] S. Dey, A. Dutta, J. I. Toledo, S. K. Ghosh, J. Lladós, and U. Pal, "Signet: Convolutional siamese network for writer independent offline signature verification," *CoRR*, 2017.

[4] M. R. Gupta, N. P. Jacobson, and E. K. Garcia, "Ocr binarization and image pre-processing for searching historical documents," *Pattern Recognition*, vol. 40, no. 2, pp. 389–397, 2007.

[5] S. F. Dodge and L. J. Karam, "Understanding how image quality affects deep neural networks," in *Eighth International Conference on Quality of Multimedia Experience, QoMEX 2016*, 2016.

[6] S. G. Dafe and S. S. Chavhan, "Optical character recognition using image processing," 2018.

[7] S. Krig, *Computer vision metrics*. Springer, 2016.

[8] I. Chakravarty, N. Mishra, M. Vatsa, R. Singh, and P. Gupta, "Offline signature recognition," in *Encyclopedia of Data Warehousing and Mining*. IGI Global, 2005, pp. 870–875.

[9] L. G. Hafemann, R. Sabourin, and L. S. Oliveira, "Offline handwritten signature verification—literature review," in *Image Processing Theory, Tools and Applications (IPTA), 2017 Seventh International Conference on*. IEEE, 2017, pp. 1–8.

[10] O. Ronneberger, P.Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, ser. LNCS, vol. 9351. Springer, 2015, pp. 234–241, (available on arXiv:1505.04597 [cs.CV]). [Online]. Available: http://lmb.informatik.uni-freiburg.de/Publications/2015/RFB15a

[11] I. B. Barbosa, M. Cristani, B. Caputo, A. Rognhaugen, and T. Theoharis, "Looking beyond appearances: Synthetic training data for deep cnns in re-identification," *Computer Vision and Image Understanding*, vol. 167, pp. 50–62, 2018.

[12] N. Pinto, D. D. Cox, and J. J. DiCarlo, "Why is real-world visual object recognition hard?" *PLoS computational biology*, vol. 4, no. 1, p. e27, 2008.

[13] Z. Obermeyer and E. J. Emanuel, "Predicting the future—big data, machine learning, and clinical medicine," *The New England journal of medicine*, vol. 375, no. 13, p. 1216, 2016.

[14] M. K. Kalera, S. Srihari, and A. Xu, "Offline signature verification and identification using distance statistics," *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 18, no. 07, pp. 1339–1360, 2004.

[15] N. Otsu, "A threshold selection method from gray-level histograms," *IEEE transactions on systems, man, and cybernetics*, vol. 9, no. 1, pp. 62–66, 1979.

[16] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, 2012, pp. 1097–1105.

[17] R. Smith, "An overview of the tesseract ocr engine," in *Document Analysis and Recognition, 2007. ICDAR 2007. Ninth International Conference on*, vol. 2. IEEE, 2007, pp. 629–633.

[18] I. Hubert, A. Arppe, J. Lachler, and E. A. Santos, "Training & quality assessment of an optical character recognition model for northern haida." in *LREC*, 2016.