2021 RIVF International Conference on Computing and Communication Technologies (RIVF) | 978-1-6654-0435-8/21/$31.00 ©2021 IEEE | DOI: 10.1109/RIVF51545.2021.9642150

MC-OCR Challenge 2021: Simple approach for receipt information extraction and quality evaluation

Cuong Manh Nguyen[1,2], Vi Van Ngo[1], Dang Duy Nguyen[1,2]
[1]*VCCorp, 19th Floor, Center building, 1 Nguyen Huy Tuong Road, Thanh Xuan District*
[2]Hanoi University of Science and Technology, 1 Dai Co Viet Road
Hanoi, Vietnam
Email: cuonghip0908@gmail.com, vingovan@admicro.vn, dangnguyenduy@vccorp.vn

***Abstract*—This challenge organized at the RIVF conference 2021 [12], with two tasks including (1) image quality assessment (IQA) of the captured receipt, and (2) key information extraction (KIE) of required fields, our team came up with a solution based on extracting image patches for task 1 and Yolov5 + VietOCR for task 2. Our solution achieved 0.149 of the RMSE score for task 1 (rank 7) and 0.219 of the CER score for task 2 (rank 1). Our code is available at https://github.com/cuongngm/RIVF2021.**

***Index Terms*—Object Detection, OCR, Image Quality Assessment, Key Information Extraction.**

I. Introduction

Recently, extracting information from document images (e.g., id card, passport, invoice, etc) and quality assessment are potential problems and have many applications (in financial, accounting, taxation areas, etc). For the quality assessment task, the difficult is that document images are generally taken by mobile devices and they may be crumpled or the content may be blurred. This results in a low quality of recognized information so this task is essential. In addition, for the task of extracting information, specifically, in this challenge of receiving, we need to classify the information fields of interest in an image this is seller, address, timestamps, and total cost.

The main contributions of this paper can be summarized as follows:

- We propose simple methods to solve the above problems. For task 1, we gathered preprocessing techniques for the input text image and used CNN to extract the information needed to measure the quality of the text image. Also for task 2, we used object detection to detect containers of information that were extracted and recognized on text images.
- Analyze, process data, and conduct experiments related to the dataset provided by the organizers.

II. Related work

A. Task 1: Image Quality Assessment (IQA)

As for task 1, we do not really know much about the methodology and documentation related to it. Towards Document Image Quality Assessment (TDIQA) [8] is a method based on text line detection to estimate document image quality, which is composed of three stages: text line detection, text line quality prediction for each line of text, and overall quality assessment. We ran into some problems trying to detect text, as we did not have enough annotations on the text line to train a detection model and using a pre-trained model was not very effective. Instead of we decided to follow by [5], a simpler approach. This method has two steps. First, the document image is divided into patches and non-informative patches are sifted out using Otsu's binarization technique. Then, quality scores are obtained for all selected patches using a Convolutional Neural Network (CNN), and the patch scores are averaged over the image to obtain the document score.

B. Task 2: Key Information Extraction (KIE)

The purpose of this task is to extract information fields such as seller, address, timestamps, and total cost instead of the total text in the image. To extract key information, several typical methods are hand-craft features based, NLP based, and graph convolution based, detailed below:

- For hand-craft features based method, the way to do is to apply predefined rules on forms, text that has a fixed layout/structure, and does not much change. Use regex, the template to match and define the corresponding fields of information. However, the most significant disadvantage of this approach is that we have to define each rule separately for each form, not adapting to a new form.
- For NLP based, the content obtained from the text box can be put into the text classification or the Name Entity Recognition (NER) [7] model to classify or identify of the entity belongs to each corresponding information field. Processing the plain text as a linear sequence result in ignoring most of valuable visual and non-sequential information (e.g., text, position, layout, and image) of the document. The advantage of this approach is the ability to adapt new data but it does not use any information about the box location, and this can lead to errors in predicting information such as total cost, because they can easily confuse fields such as unit price, refund if location information is not available.
- For graph convolution based method, it has many advantages and effectiveness compared to the previous two methods. It achieves the state of the art results. In the SROIE dataset of the ICDAR 2019 [3] competition, two models based on graph convolution, LayoutLM [13] and PICK [14] have high position in the rankings. It improves

extraction ability by automatically making full use of the textual and visual features within documents.

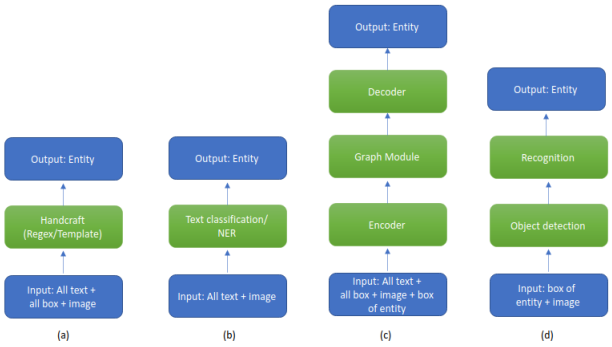

Fig. 1. Typical architectures and our method.(a) hand-craft features based method. (b) NLP based method. (c) Graph convolution based method. (d) our proposed method

A common of these methods is that it is necessary to first solve two sub-problems: text detection and text recognition. In the SROIE [3] dataset, the annotations have the coordinates of the box and the text of all the text in the image, which makes the training model easier, while in this competition, the coordinates of box are only in the extracted fields.
We surveyed and found that, in the dataset provided, receipts with a similar structure account for the majority, and a few exceptions. We think it is valid to apply an object detection model to detect text fields as objects. We tried using the yolov5 model to detect four types of seller, address, timestamps, total cost and got a good results.

III. Methodology

A. Task 1: Image Quality Assessment

We preprocess a grayscale document image with local normalization, crop the image into patches, use the CNN to estimate quality scores for selected patches, and average the scores to obtain a score for the image.

a) Algorithms: The method of implementation is as follows:

Preprocessing: As in [9], we crop the receipt image by line segment detector and perform a local normalization over the entire image. Each pixel is subtracted by the mean and divided by standard deviation of the pixels in a surrounding window. Fig.2(a) and (b) show a document image after crop and its local normalization result.

Patch Sifting: We perform Otsu's binarization [10] on the raw image, and obtain a binary map corresponding to foreground and back-ground. We crop patches from the pre-processed (i.e. locally normalized) images and check their corresponding patches on the binary map. If the patch on the binary map is constant, i.e.all ones or all zeros, then this patch is discarded. Since the patch size is chosen to be larger than the typical stroke width, text patches are preserved. Most patches sifted out in this way are background patches or non-text foreground patches. Fig.1(c) shows the locations of patches that are selected after sifting.

Fig. 2. (a) A receipt scanned (b) local normalization result (c) mask of non-constant 48x48 patches(white)

Network Architecture: Once the patches are obtained, we feed them into a network. Fig.3 shows the architecture of the proposed network. The network contains two convolution and pooling layers, two fully connected layers and one output layer. The input is sifted patches of size 48×48. The first convolution layer contains 40 kernels each of size 5×5, followed by a 4×4 max pooling, then the second convolution layer with 80 kernels each of size 5×5. Following the second convolution layer is a special max-min pooling that we will explain later. Each of the two fully connected layers contains 1024 nodes. The last layer is a linear regression that outputs the quality score.

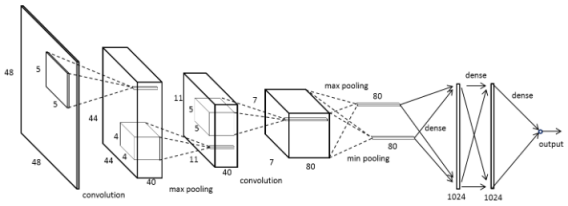

Fig. 3. The architecture of model

b) Parameter settings and processing time: We splited the dataset according to train/ val/ test ratios of 0.7/ 0.2/ 0.1 respectively, and used a batch size of 32. We used the Adam optimizer [6] and the loss function is the L1 loss. Detailed descriptions can be found in the configuration file of the source code. The training time for runs of models was approximately one hours on the server GTX 2080Ti.

B. Task 2: Key Information Extraction

This task aims to detect and recognize information about the seller, address, timestamps, and total cost in the receipt image.

a) Different techniques attempted: We wanted to used a model based on graph convolution (PICK) [14] for this task, but we had difficulty handling the input data. Input of this model required the coordinate boxes and transcripts of all text in the image + entities containing information of output fields. Based on the dataset provided, we only have entities that contain information about field, address, timestamp, and total cost. We need to do the detection and recognition task based on public pre-trained models. For text detection, we used the pre-trained CRAFT [1]. The output of the CRAFT model is a binary image representing the proposed area that appear to contain text. We did some post-processing methods to expand the proposed area, then cropped and aligned to get the text box. Fig.4 show some results after used CRAFT.

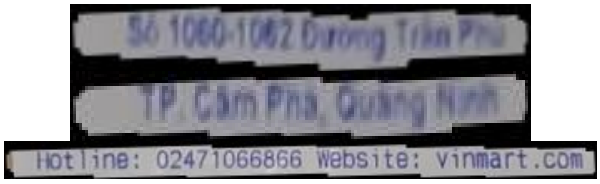

Fig. 4. Cropped images after used CRAFT

Then we recognized these images using the pre-trained VietOCR model. However, these results are not a good input to the PICK model. The results after training 50 epochs using the PICK model (with default settings) are obtained in the table:

TABLE I
RESULTS AFTER TRAINING THE PICK MODEL

Name	*mEP*	*mER*	*mEF)*
total cost	0.842	0.814	0.827
address	0.724	0.94	0.818
timestamp	0.867	0.838	0.852
seller	0.832	0.862	0.847
overall	0.815	0.845	0.83

The evaluation metric are mean entity precision (mEP), mean entity recall (mER), mean entity F1-score (mEF). When submitting the result file for the private test dataset, the CER score of the model using the PICK model worse than our final solutions. We assume the reason is due to the training data being built from other pre-trained models instead of using human-annotations.

b) The final solution: We found that the data has a similar structure in the dataset, so we tried to apply a object detection model (yolov5) [4] to detect four classes of seller, address, timestamp, and total cost. Then, we recognized the text in the boxes images using the VietOCR library.

Yolov5: is a powerful and outstanding current object detection pattern. Its source code is very clean and easy to access, its author applies a lot of tricks and turtorials in the public repository on github, such as augmentation, ensemble model, test time augmentation, weights box fusion, etc.

In this competition, we received the dataset from the organizer then converted it into yolov5 format to training model. Then apply the relevant experiments (increase in size, augment, ensemble, etc.) to get the optimal model. In training phase, we used all the train dataset provided by the organizers and used the warm up data as a validation set, the image size used was 1024 and applied augmentation (flipud, fliplr, resize,...). After training with two pre-trained models: yolov5s and yolov5m, we ensembled them together and got the final model with smaller losses and better accuracy.

Fig. 5. Some training results

VietOCR: VietOCR is a very effective Vietnamese language recognition library recently. Model architecture is a great combination between the model CNN and Transformer [11] (which is the foundation model of BERT [2] quite famous). The author installs both the sequence model, the attention seq2seq and the transformer.

In this competition, we trained model both the attention OCR and the transformer OCR, evaluate the result and submit the model with better character error rate.

In the dataset provided by the organizers, there are a lot of receipt images rotated 90, 180, 270 degrees. This causes a lot of problems for us, the solution we came up with was to rotate the input image to four directions to produce four images and make a predict based on each of them and pick out the best results. Besides, we also applied thresholds during the detection and recognition phase.

IV. ANALYSIS

In this chapter, we will present and compare the results obtained from related experiments.

A. Task 1: Image Quality Assessment

We did not really focus our work on this task, but rather on task 2. In this task, after more than one hour of training, we obtained a model with the size of 5.2 MB and made a prediction on the test dataset. The inference time per image was approximately 400-600ms. The submit result achieved the RMSE score was 0.149 on the private test set.

B. Task 2: Key Information Extraction

In detection phase, before, we tested the training based on pre-trained model yolov5s with two input sizes (640 and 1024) in the same all other parameters. The results are obtained in the table:

The evaluation metric is mean average precision (mAP). It can be concluded that the training model with a large input size and similar to the size of the dataset will improve the accuracy of the model and the loss function converged

TABLE II
RESULTS COMPARISON OF PRE-TRAINED WITH TWO INPUT SIZE

Input size	*Total loss*	*mAP(:.5)*	*mAP(.5:.95)*
640	0.0567	0.932	0.673
1024	0.0525	0.946	0.728

better. Next, we experimented with some pre-trained models (yolov5s, yolov5m) with the input size of the image equal 1024.

TABLE III
TRAINING WITH THE INPUT SIZE 1024 AND DIFFERENT PRE-TRAINED YOLOV5 MODELS

Pre-trained	*Model size*	*Train time*	*Total loss*	*mAP(:.5)*
yolov5s	15MB	50'	0.0525	0.946
yolov5m	41MB	1h25'	0.0528	0.938
ensemble	-	-	0.0522	0.951

Usually the training with the large model of yolov5 will be more accurate, along with the more resource consuming, in this case we think that using the pre-trained yolov5s, yolov5m is suitable for balancing the above factors.

In recognition phase, we experimented with both the seq2seq and transformer OCR model.

TABLE IV
RECOGNITION

Model	*Size*	*infer*	*acc full seq*	*acc per char*
vgg-transformer	151.9MB	84ms	0.876	0.704
vgg-seq2seq	89.6MB	11ms	0.871	0.684

During the inference, the vgg-seq2seq model proved to be superior in terms of speed, while its accuracy was only slightly worse. When submitting the result file to the private test dataset, the CER score of the model using the vgg-transformer model is 0.217 and the vgg-seq2seq model is 0.219. Our final model predicts with time 500-700ms per image, statistics obtained when running on a server GTX 2080Ti.

LESSONS LEARNED

This contest is our first time trying. Although it was a bit of a surprise, we tried to complete our exam well, and this was also the place where we met and learned a lot from the solutions of other teams. The competition provided with access to a variety of receipt datasets, a valuable resource for everyone to research together. The competition helped us improve coding, visualize data, process data, noise filtering, and some object detection tricks.

We are waiting for the competition of a more complete dataset (with a format similar to the SROIE dataset) so that teams can come up with better and better solutions (including my team).

CONCLUSION

Summary

In this article, we presented our methodology and understanding in completing the challenge of the competition. Sincerely thank the organizers for creating the exciting contest and their enthusiastic support for the teams.

Potential future work

We understand that our solution is still limited and will continually research and improve current results. Besides, learning solutions from other teams is also what we will do.

REFERENCES

[1] Youngmin Baek, Bado Lee, Dongyoon Han, Sangdoo Yun, and Hwalsuk Lee. Character region awareness for text detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 9365–9374, 2019.

[2] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805, 2018.

[3] Zheng Huang, Kai Chen, Jianhua He, Xiang Bai, Dimosthenis Karatzas, Shijian Lu, and C. V. Jawahar. Icdar2019 competition on scanned receipt ocr and information extraction. *2019 International Conference on Document Analysis and Recognition (ICDAR)*, Sep 2019.

[4] Glenn Jocher, Alex Stoken, Jirka Borovec, NanoCode012, ChristopherSTAN, Liu Changyu, Laughing, tkianai, Adam Hogan, lorenzomammana, yxNONG, AlexWang1900, Laurentiu Diaconu, Marc, wanghaoyang0106, ml5ah, Doug, Francisco Ingham, Frederik, Guilhen, Hatovix, Jake Poznanski, Jiacong Fang, Lijun Yu , changyu98, Mingyu Wang, Naman Gupta, Osama Akhtar, PetrDvoracek, and Prashant Rai. ultralytics/yolov5: v3.1 - Bug Fixes and Performance Improvements, October 2020.

[5] Le Kang, Peng Ye, Yi Li, and David Doermann. A deep learning approach to document image quality assessment. In *2014 IEEE International Conference on Image Processing (ICIP)*, pages 2570–2574, 2014.

[6] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization, 2017.

[7] Guillaume Lample, Miguel Ballesteros, Sandeep Subramanian, Kazuya Kawakami, and Chris Dyer. Neural architectures for named entity recognition. *CoRR*, abs/1603.01360, 2016.

[8] Hongyu Li, Fan Zhu, and Junhua Qiu. Towards document image quality assessment: A text line based framework and a synthetic text line image dataset, 2019.

[9] Anish Mittal, Anush Krishna Moorthy, and Alan Conrad Bovik. Noreference image quality assessment in the spatial domain. *IEEE Transactions on Image Processing*, 21(12):4695–4708, 2012.

[10] Nobuyuki Otsu. A threshold selection method from gray-level histograms. *IEEE Transactions on Systems, Man, and Cybernetics*, 9(1):62–66, 1979.

[11] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. *CoRR*, abs/1706.03762, 2017.

[12] Xuan-Son Vu, Quang-Anh Bui, Nhu-Van Nguyen, Thi-Tuyet-Hai Nguyen, and Thanh Vu. Mc-ocr challenge: Mobile-captured image document recognition for vietnamese receipts. In *Proceedings of the 15th IEEE-RIVF International Conference on Computing and Communication Technologies*, RIVF '21. IEEE, 2021.

[13] Yiheng Xu, Minghao Li, Lei Cui, Shaohan Huang, Furu Wei, and Ming Zhou. Layoutlm: Pre-training of text and layout for document image understanding. *CoRR*, abs/1912.13318, 2019.

[14] Wenwen Yu, Ning Lu, Xianbiao Qi, Ping Gong, and Rong Xiao. Pick: Processing key information extraction from documents using improved graph learning-convolutional networks, 2020.