An End-to-End Method to Extract Information from Vietnamese ID Card Images

Khanh Nguyen-Trong
Department of Software Engineering
Posts and Telecommunications Institute of Technology
Hanoi, Vietnam

***Abstract*—Information extraction from ID cards plays an important role in many daily activities, such as legal, banking, insurance, or health services. However, in many developing countries, such as Vietnam, it is mostly carried out manually, which is time-consuming, tedious, and may be prone to errors. Therefore, in this paper, we propose an end-to-end method to extract information from Vietnamese ID card images. The proposed method contains three steps with four neural networks and two image processing techniques, including U-Net, VGG16, Contour detection, and Hough transformation to pre-process input card images, CRAFT, and Rebia neural network for Optical Character Recognition, and Levenshtein distance and regular expression to post-process extracted information. In addition, a dataset, including 3.256 Vietnamese ID cards, 400k manual annotated text, and more than 500k synthetic text, was built for verifying our methods. The results of an empirical experiment conducted on our self-collected dataset indicate that the proposed method achieves a high accuracy of 94%, 99.5%, and 98.3% for card segmentation, classification, and text recognition.**

***Keywords*—Optical character recognition; U-Net network; VGG16 network; CRAFT network; rebia network**

I. Introduction

The ID card is the most widely used identity document of Vietnamese citizens. It provides crucial information used for many business processes, such as ID number, name, address, and date of birth. However, extracting information from such cards is usually carried out manually, which is time-consuming, tedious, and can prone to errors. In this context, methods that automatically analyze and extract information, such as Optical Character Recognition (OCR) are frequently used.

However, there have been several challenges in reading information from cards captured in natural scenes, including difficulties in scene text recognition, lacking training data, and the complexity of Vietnamese language. According to *et al.* [1], the diversity of scene text, the complexity of the background, and the interference factors are the most difficulties for scene text detection and recognition. The first difficulty is caused by diversities in fonts, colors, scales, and text orientations. For example, a Vietnamese ID card can contain three or four different fonts and colors. Moreover, there are four types of cards, as illustrated in Fig. 1, which have several formats for the same field. Especially for the 9-digit ID card, it can contain handwriting text or old fonts created by typewriters. The complexity of background leads to difficulty to clearly distinguish texts from backgrounds. For instance, the Vietnamese ID card background is usually incorrectly detected as text. As shown in Fig. 2a, the bounding box, the detected text region, also contains the pattern background. The interference factors, such as noise, blur, distortion, low resolution, non-uniform illumination, and partial occlusion make the detection and recognition harder, as illustrated in 2b.

Due to sensitive information, researchers have faced many difficulties in collecting Vietnamese ID cards for model training. This lack of data can easily lead to bad results in extracting information. Moreover, the Vietnamese alphabet is a Latin-based alphabet, but with many additional characters, including five accent symbols (*á, à, ả, ã, ạ*) and derivative characters, such as *ê, ă, ử* ... [2]. Therefore, we cannot apply available OCR methods and pre-trained models, usually for English, to this language.

There have already been a few studies on similar cards, for example, Egyptian ID [3], Indonesian ID [4], or even Vietnamese ID cards [5], [6]. Most of them are deep learning-based due to the outstanding performance in OCR [7]–[9]. However, the proposed methods focus either on different languages, like English or on sub-tasks, such as text recognition [5], [6]. They cannot be used for the Vietnamese language or to directly deal with raw images captured in natural scenes.

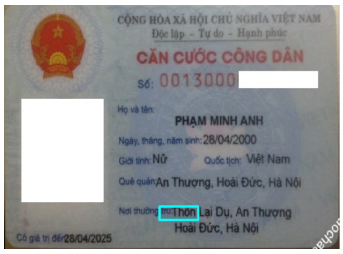
Therefore, this paper presents an end-to-end method for information retrieval from Vietnamese ID card images. The method consists of 3 consecutive steps (Pre-processing, Tex detection and recognition, and Post-processing) with four neural networks, two image processing techniques, and two basic Natural language processing to deal with raw card images captured in natural scenes. We also created four datasets to deal with the lack of data. In summary, the major contributions of this work are as follows::

- We proposed an end-to-end deep learning-based method to extract information from the Vietnamese ID card, which is based on state-of-the-art methods in related fields.
- We introduced four neural networks (U-NET, VGG16, CRAFT, and Rebia) to analyze the card and extract its content. To assure the correlation among models, we based on the output of the previous step to train models for the subsequent one.
- We applied (i) Contour detection and Hough transformation at the pre-processing step to align and crop the card; (ii) Levenshtein distance and regular expression at the post-processing step to correct extracted information.

(a) 9-digit ID Card (b) 12-digit ID Card (c) New 12-digit ID Card (d) Chip-based ID Card

Fig. 1. Front Side of Vietnamese ID Card.

(a) Pattern background as text



(b) Occlusion text

Fig. 2. Complexity of Background and Interference Factors.

- We built four datasets for Vietnamese OCR, including two manual (ID Cards and manually annotated text) and two synthetic (synthetic image and text) datasets. The datasets contain more than 400k manual labeled text from 3.256 Vietnamese ID Card. Furthermore, we evaluated our proposed method on these datasets and highlighted the experimental results thus obtained.
- We proposed a microservice architecture to deploy our method on a real system, which allows balancing the information flow between each step of the end-to-end method.

The remainder of this paper is structured as follows. Section 2 discusses relevant previous studies. Section 3 provides the details regarding our proposed networks. The experimental evaluation is presented in Section 4, and finally, some concluding remarks and a brief discussion are provided in Section 5.

II. Related Works

In general, information retrieval from ID card images relates to the OCR of semi-structured documents, such as receipts [10], bank cards [11], business cards [12], [13], invoices [14], [15], and so on. It typically contains several common steps, including pre-processing, text detection, text recognition, and layout analysis [5], [10], [16]–[18].

Clearly, in the context of Scene Text Recognition (STR), where document images can be affected by many factors, pre-processing is necessary to improve the quality and normalize the input data. This step can include a series of sub-tasks, such as document detection, segmentation, alignment, and basic image processing techniques. For example, with a raw image captured in natural scenes, we must check the existence of documents, and then extract their position. The latter usually produces information about the top-left and bottom-up corners of the box that covers the object. To have a stable result in further steps, the document can be aligned vertically, in which the text direction is from left to right.

The output of pre-processing is passed to the next step, where the text detection is essentially performed [5], [16]. Then, at the layout analysis step, these ROIs are extracted and classified into corresponding fields, for example, ID number, name, date of birth, address. At the final step, where the main principle of what is usually called optical character recognition happens, we predict the potential string of extracted text areas.

Regarding the order of these steps, the pre-processing is usually performed first, while the remaining can be varied. For example, text detection and recognition can be performed together, as in the stepwise methodologies or step-by-step like integrated methodologies [19]. The layout analysis is typically done after recognizing, which determines the corresponding label of ROI and so on.

Recently, with the development of deep learning, many methods have been proposed for these steps with potential accuracy. For the pre-processing step, they usually applied both traditional image processing techniques [20] and deep neural networks such as the canny edge detection algorithm [10], Otsu's method [5], U-Net [21], and VGG [22]. Applying deep learning to OCR also achieved higher performance and low processing time than traditional machine learning. Regarding text detection, for example, Baek *et al.* [23] presented a character-based method, Character Region Awareness for Text Detection (CRAFT) that effectively detects text areas by exploring each character region and affinity between them. Liao *et al.* [24] proposed another method that can detect the character in real-time. Similarity these are also many models in the literature for text recognition, such as CHAR model [25], CTPN [26], TRBA [27], Shi *et al.* [28] and so on. They can be categorized into character-level and word-level. The first approach firstly locates the position of each character, then

recognizes them by a classifier, and groups characters into the final text. Meanwhile, the second method, which outperforms the first one, considers the text line as a whole and focuses on mapping the entire text into a target string sequence [29].

These methods are applicable to many language types, such as English, Korean, Chinese. However, it is impossible to directly apply to Vietnamese that contain additional accent and diacritical marks. Additional training is needed to learn its specific features. Moreover, they usually focus on sub-steps or a specific problem, e.g., only pre-processing [5], or text detection [23]. We cannot simply juxtapose these steps for an end-to-end method to extract information from the Vietnamese ID card. To achieve high and stable performance, they need to correlate with each other. The output of the previous step should be used as the input to train models for the subsequent step.

III. Proposed Method

Due to the variety of captured images and types of ID cards, the proposed method contained three steps with four neural networks, as shown in Fig. 3. It consisted of a set of deep learning and traditional machine learning techniques in Computer Vision and Natural Language Processing, as follows:

- We performed several pre-processing steps, including segmentation, alignment, and identification, to determine and normalize the card. First, we detected and segmented cards from input images. Next, they were vertically aligned and cropped, with text from left to right. Then, we determined their type (9-digit, 12-digit, or the new 12-digit ID Card). Two deep learning methods, and two image processing algorithms were applied for this step: U-Net model for the detection, and segmentation, VGG16 model for the classification, Contour Detection, and Hough Transformation for the alignment.
- We applied the word-level approach to detect and recognize Vietnamese optical texts on the cards, including the CRAFT method [23] for text detection, the Attn method with ResNet, and BiLSTM for text recognition.
- Lastly, to correct text errors and identify text fields (Named Entity Recognition), such as names, date of births, we performed two main tasks, including layout analysis and text correction. The Levenshtein distance, regular expression, and two pre-defined dictionaries were applied.

The input of the first step is images containing Vietnamese ID cards, while the output is the ID cards that were cropped, aligned from the background. The type of card is an important output of this step. Then, the cropped ID is used as input for the next step (text detection and text recognition), which produces two lists: a list of predicted texts, and a list of bounding boxes. The last step takes these lists and the type of cards, from the first step, to analyze and results in a list of texts with their field. More detail description of these steps will be presented in the following subsections.

A. Preprocessing: Segmentation, Normalization, and Identification

Owing to the unconstrained nature of captured images, three preprocessing steps were applied to make them available for the next steps, including segmentation, normalization, identification. These tasks allowed us to separate the cards from background images, vertically align them, and identify their type.

Thanks to its efficiency in image segmentation, we trained a U-Net model to segment Vietnamese ID cards. The network has the same architecture as the work of Ronneberge *et al.* [30]. But, instead of using 512 x 512 input images, we downscaled to 256 x 256 pixels. The output was a 128 x 128 image that is used then as a mask to segment the card.

The model allows us to determine a binary mask of cards. Thus, we combined two basic image-processing techniques to align and crop cards: Contour Detection, and Hough Transformation. The first algorithm was applied to the binary image to detect the boundaries of cards. Then, we transformed these lines to Hough coordinate to find two interacted parallel lines that bound the card. It allows us to determine four corner coordinates of the card. From these corners, we applied a perspective transformation to align and crop the card to 600x400 pixel images.

Lastly, we identified the type of cropped cards by fine-tuning the VGG16 network trained on ImageNet [31]. This network consisted of sixteen layers: thirteen convolutional and three fully connected layers. Each layer contained convolutional layers, max-pooling layers, and fully connected layers. We fine-tuned the model with our dataset to classify eight classes: the front, the back, the reversed front, and back of the 9-digit, 12-digit cards. Therefore, the feature extractor of VGG16 was kept as the original network. To adapt to our dataset, we updated the classifier section with a new fully-connected layer that adjusts to 8 classes only.

B. Text Detection and Recognition

Vietnamese is a Latin-based language that has several additional accents and diacritical marks [32]. The language has more than 250 characters. Among twenty-nine alphabetic scripts, the language consists of twenty-two Latin letters ('f', 'j', 'w', and 'z' letters are eliminated). The remaining are created by combining these letters with diacritics located just at the top or bottom of letters, with or without a small gap between them.

Therefore, instead of training a new model which requires many manual labeled datasets, we adapted models trained on English datasets to detect text. Thanks to its performance in dealing with the low-quality dataset, we applied the CRAFT text detector to localize the text in cropped cards. The model supports *effectively detect text area by exploring each character and affinity between characters* [23].

CRAFT has three detection levels: (i) individual character; (ii) individual word; and (iii) connected words or sentences. The processing time of the first level is long, while the third level is unstable. Thus, we applied only the second level to detect individual words on the cards.

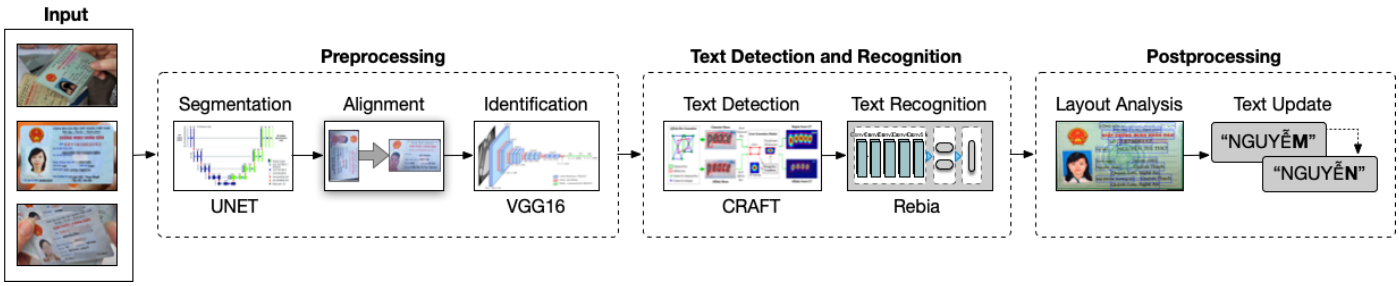


Fig. 3. Information Retrieval from Vietnamese ID Card Images.

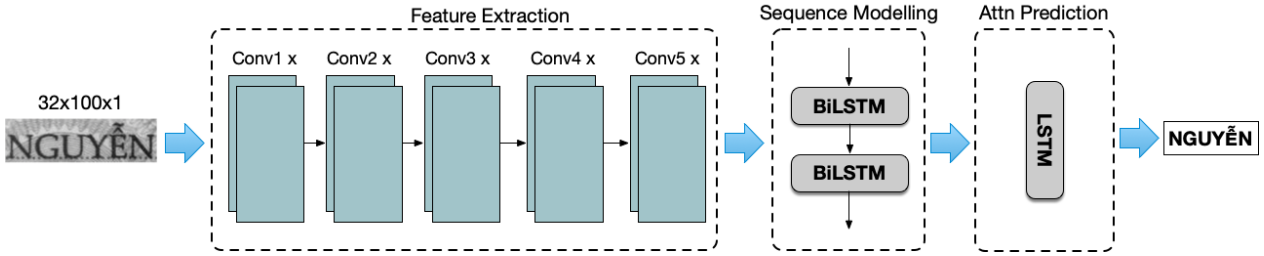


Fig. 4. Rebia Neural Network for Text Recognition.

The model allows detecting only the alphabet word without the tone marks, such as the grave accent, hook above, tilde, acute accent, and dot below. Therefore, we adapted the detector to cover the tone marks by enlarging bounding boxes of detected areas.

The input of this step is 600x400 pixel images that were segmented and aligned previously. The output is a list of bounding boxes that contains cropped words.

For text recognition, we propose a deep neural network, namely Rebia, to recognize the optical words on Vietnamese ID cards. Unlike most existing methods in the field, we didn't apply the rectification step before extracting the feature. Since these texts have a similar orientation and shape, rectification, which is used to normalize different types of texts (i.e., curved and tilted texts), is not necessary. The proposed network contained three blocks, as shown in Fig. 4 and detailed in Table I. First, the feature map that focuses on the word level was extracted by ResNet neural network. The network contained five layers. We converted all text images to gray-scale of size 100 x 32 pixels, to normalize the input data.

Next, we reshaped the extracted features to a sequence feature used for prediction. Thanks to its capability of capturing contextual information within a sequence [33], we used two Bidirectional Long Short-Term Memory (BiLSTM) at this step. The two networks have the same hidden unit that is 256.

Lastly, an attention-based decoder was employed to predict the sequence feature. It contained an LSTM layer with 256 hidden units.

We applied the ReLu activation for three blocks. After each convolution, a batch normalization was used to standardize its outputs. The objective function was the negative log-likelihood of the probability of label sequence.

C. Post-processing: Layout Analysis and Text Update

After recognizing, we categorized the text to the corresponding fields, such as the name, ID, address, and date of birth. Due to the lack of a training dataset, we combined natural language processing, regular expression techniques, and dedicated layout analysis algorithms to determine field types.

The front of a Vietnamese ID card is typically organized from left to right, top to bottom, and line by line, while the back is more complicated. But for the back, we are interested only in the issued date and place, which has the same structure. Therefore, we first sorted bounding boxes of detected words from left to right and line by line.

Next, we evaluated and updated the recognized text by the Levenshtein distance with the help of several domain-specific dictionaries. We combined these texts with results from the above step to identify different fields. Furthermore, for fixed-format fields, such as the date, number, and ID, we also applied the regular expression and algorithms, as follows:

- For the ID number: By experiment, we found that one of the most common issues for this field was the overlap of the caption and content (ID number). It made the recognized ID numbers have several additional characters at the beginning. Therefore, for a text line that was determined as the ID number, we took only a fixed number of digits, starting from the end of the text (9 for the old card, 12 for the new card).
- For the date of birth, expired, and issued date: Each card type has its format to represent the date, so we applied a rule-based mechanism and the regular expression to check the text.

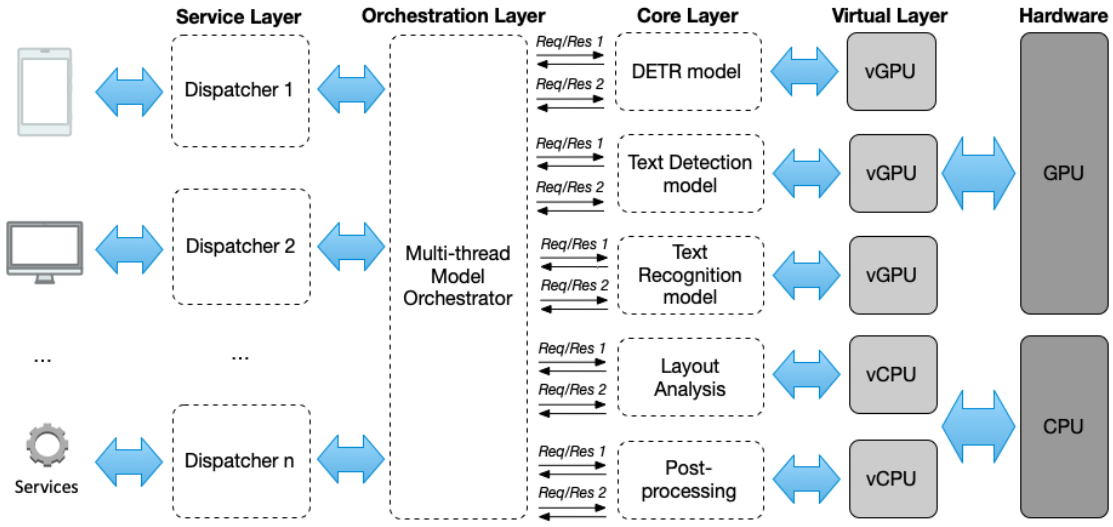


Fig. 5. Multi-layer Architecture for Load Balancing of Model Execution.

TABLE I. DETAILED ARCHITECTURE OF REBIA FOR TEXT RECOGNITION

Feature Extraction - Resnet		
Layer	Configuration (kernel, stride, padding, channel)	Output
conv1 x	Conv1: 3x3, 1x1, 1x1, 32 Conv2: 3x3, 1x1, 1x1, 64 MaxPool: 2x2, 2x2, 0x0	100x32
conv2 x	01 BasicBlock: 3x3,128 3x3,128 Conv3: 3x3,1x1,1x1,128 MaxPool: 2x2, 2x2, 0x0	50x16
conv3 x	02 BasicBlock: 3x3, 256 3x3, 256 Conv4:3x3, 1x1,1x1,256 MaxPool:2x2, 2x2, 0x0	25x18
conv4 x	05 BasicBlock: 3x3, 512 3x3, 512 Conv5:3x3, 1x1,1x1,512	26x4
conv5 x	03 BasicBlock: 3x3,512 3x3,512 Conv6:2x2,1x2,1x0,512 Conv7:2x2,1x1,0x0, 512	26x1
Sequence Modelling - BiLSTM		
BiLSTM	Hidden units:256	256
BiLSTM	Hidden units:256	256
Prediction - Attn decoded		
LSTM	Hidden units:256	256

- For the name and address: First, we built three dictionaries containing common Vietnamese family names, middle names, and addresses. Then we applied the Levenshtein algorithm and regular expression to correct the text, if necessary.

D. Multi-layer Architecture for Load Balance of Model Execution

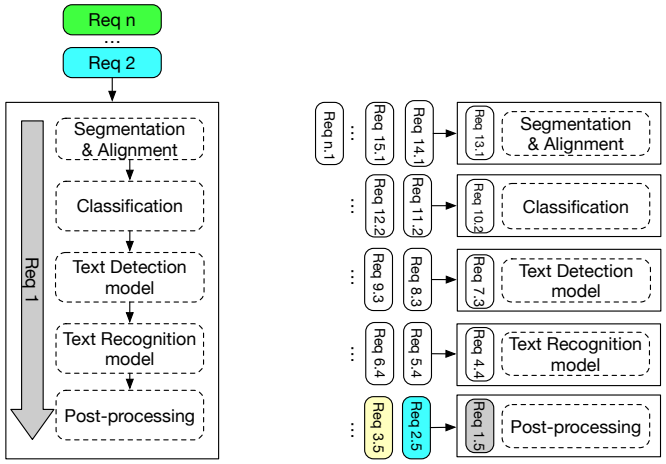
In this study, we propose a multi-layer system architecture to deploy the end-to-end method, which supports load balancing of model execution. We considered each model as an independent and parallel process, which can serve different ID cards at the same time, as shown in Fig. 6. It eliminated the bottleneck at some models due to high memory consumption, especially text detection and recognition.

Therefore, we deployed the proposed end-to-end method into five sub-tasks and processed them separately. Each one consists of a model/algorithm with different input and output. Owing to model orchestrators and API gateways, the proposed architecture can coordinate the input and output of these sub-tasks, as presented in 5.

Thus, each information extraction request was split into five sub-requests, as shown in Fig. 6. On the left figure (single request invocation), all models are blocked until the last sub-task (post-processing) finishes, while on the right (multi-request supports), models are free if they finish their jobs. (A request n was split into five sub-requests (*Req n.x*); x denotes the sub-task that corresponds to different steps of the proposed method.)

We applied the FIFO (First In, First Out) technique to handle multi-sub-requests. The complete architecture of our system is presented in Fig. 5, which contains three layers and is based on the virtual technique to maximize infrastructure using, as follows:

- The orchestration layer contains the model orchestrators that are responsible to coordinate different steps.
- The gateway layer includes different API gateways that interact with the corresponding micro-services et each step.
- The micro-services layer is composed of different micro-services that support API to interact with models.



(a) Single request end-to-end invocation (b) Multi-requests end-to-end invocation

Fig. 6. Single (a) and Multi Request (b) End-to-end Services.

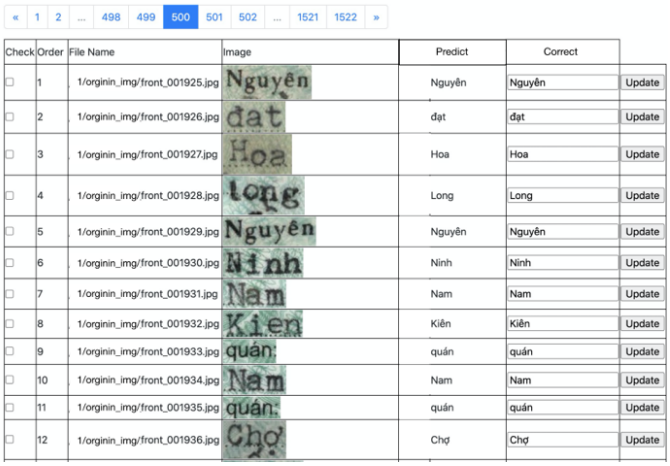


Fig. 7. Dedicated Annotation Tool.

- The infrastructure layer applied virtual techniques to share physical hardware (GPU, CPU, and different virtual GPU and CPU) among micro-services.

IV. Experiments

A. Dataset

We used four datasets in this study, including two manual (ID Cards and manually annotated text) and two synthetic (synthetic image and text) datasets for training U-Net, VGG16, and Rebia. The following steps were performed to prepare our datasets:

- **ID cards**: we collected 3.256 Vietnamese ID cards from volunteers and public images on the internet. It contains 1.530 samples for the 9-digit card, 935 for the 12-digit card, and 783 for the new 12-digit (since the chip-based ID card has just been released in 2021, in this study we focused only on the three other cards). We then used the labelImg[1] tool to annotate this dataset.
- **Synthetic images**: we extracted ID cards from the above dataset and randomly put them on background images containing different objects (i.e., papers, business cards, license cards, and so on). We thus generated a total of 60k synthetic images.
- **Manually annotated text**: We first applied the CRAFT model on extracted ID cards to detect individual words. Next, Tesseract OCR was applied to predict the text. Then, we developed a dedicated tool to correct the texts manually, as shown in Fig. 7. This process is illustrated as in Fig. 8, phase 3. Lastly, we obtained a total of 400k manually annotated texts.
- **Synthetic text**: This dataset consists of more than 500k synthetic texts generated from popular Vietnamese names, addresses, numbers, etc. We used a tool, namely Synthetic Data Generator[2] to generate this dataset.

For the manually annotated and synthetic text, we also applied data augmentation methods to balance and increase samples, including rotation, blur, noise, and so on. Finally, we obtained a total of 8M samples. The first two datasets were used to train U-Net and VGG16, while the last two datasets were applied to train Rebia.

B. Model Training Setup

The proposed method is a continuous process, in which the output of the previous step is the input of the subsequent step. The used models should correlate with each other. Therefore, we based on the previous model to train the next model.

As detailed in Fig. 8, the model training contains three phases:

1) *Phase 1 - Segmentation model training*: we trained and validated the U-Net model on ID cards and synthetic image datasets.
2) *Phase 2 - Classification model training*: we used the trained U-NET model to extract ID cards from the first dataset and vertically align them. To preserve consistency between the segmentation and identification step, we combined these cards with those manually extracted from the same dataset to train the VGG16 model.
3) *Phase3 - Recognition model training*: similarly, we used the trained VGG16 and CRAFT model to detect and crop text areas. Based on these areas, we created the manually annotated text (as presented in the previous subsection). This dataset was then combined with the systemic text to train the Rebia model.

All models were implemented using the TensorFlow Framework 2.3.0 and Python 3.6.9, on an NVIDIA Tesla K80 GPU with a 12 GB memory and an Intel(R) 2.3Ghz

[1]https://github.com/tzutalin/labelImg

[2]https://github.com/Belval/TextRecognitionDataGenerator

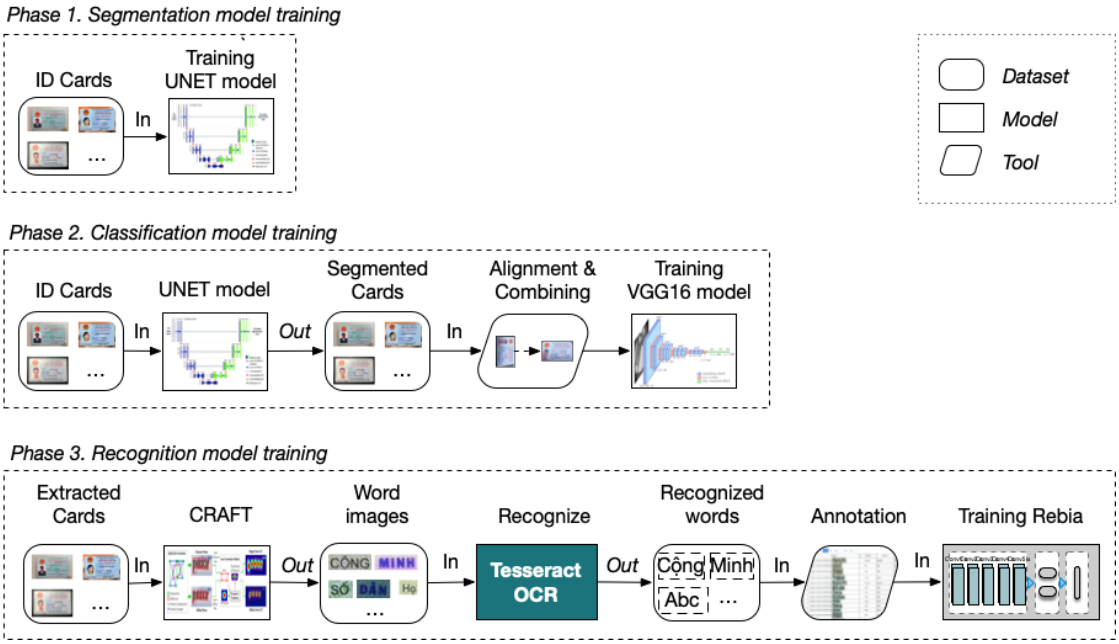


Fig. 8. Model Training Setup.

Xeon(R) micro-processor. We used the following parameters and techniques to train the models:

- To prevent bias, we divided all datasets into two subsets for training (80% randomly sample) and testing (20% randomly sample).
- For the loss function, we used a binary cross-entropy, categorical cross-entropy, and attention loss function to train the U-NET, VGG16, and Rebia models, respectively.
- For the optimization, we applied an Adam optimizer with $\beta_1 = 0.9$, $\beta_2 = 0.999$, and $e = 10^{-7}$. The initiated learning rate was 10^{-4}, and a self-adjusting learning rate technique (lr) to train U-NET and VGG16; while, Rebia was used an Adadelta optimizer with $\beta_1 = 0.9$, $rho = 0.95$, $eps = 10^{-8}$, and $lr = 1$.
- To minimize the cost function, we applied a mini-batch with a size of 192 for Rebia, 128 for U-NET and VGG16.
- The early stop technique was employed to increase the training speed and reduce over-fittings. It makes the three models stop learning if they have reached their maximum accuracy.
- The shuffle data were used, such that the models could learn randomly and provide more objective results. Before selecting the batches, we conducted a shuffling process to balance the dataset, in which fifty percent of samples were randomly chosen from each one.

After successfully training and validating the models, we deployed the end-to-end method on the same hardware configuration. To support the proposed multi-layer architecture, we implemented four Docker containers. Instance segmentation, text detection, and text recognition were hosted by three separated dockers, while both layout analysis and post-processing were hosted by only one docker.

V. Results and Discussion

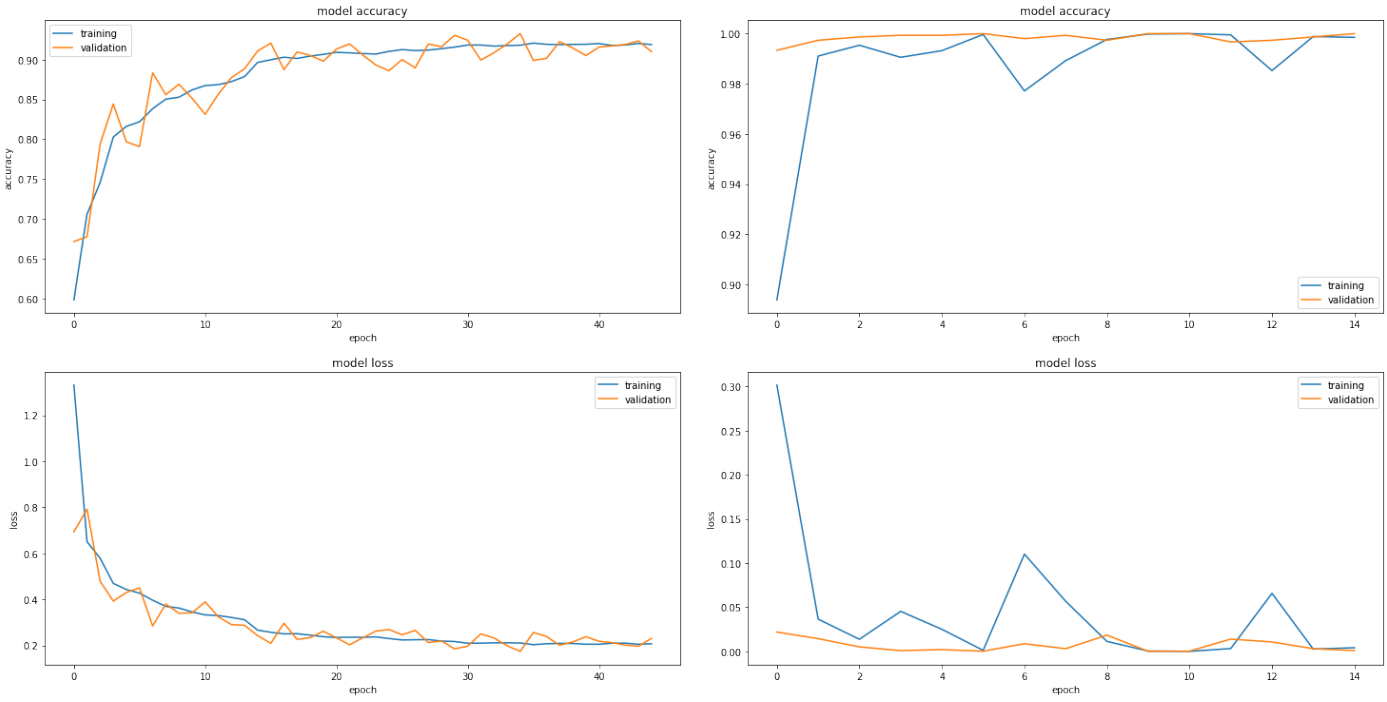
Owing to the early stop technique, the U-NET and VGG16 training were stopped after 48 and 15 epochs, as shown in Fig. 9a and 9b. The two figures also show that the gap between training loss and test loss is tiny, which means that the model operated accurately, without any overfitting. The accuracy of U-NET and VGG16 is 94% and 99.5% on the test set, respectively, as shown in Table II. For the Rebia model, the training was converged at 60 epochs. The model achieves a high accuracy of 98.3% on the validation test, as shown in Fig. 11.

TABLE II. Accuracy of the Proposed Models on the Testing Set

Model	Accuracy
U-NET (Segmentation model)	94.0%
VGG16 (Classification model)	99.5%
Rebia (Recognition model)	98.3%

Thanks to transfer learning, U-Net, and VGG16 models quickly reached a high accuracy after several initial steps/iterations. They improved only 2 – 5% of accuracy during the remaining time. It can be explained by the fact that the pre-trained models were trained on a large dataset (the ISBI dataset for U-NET, ImageNet for VGG16).

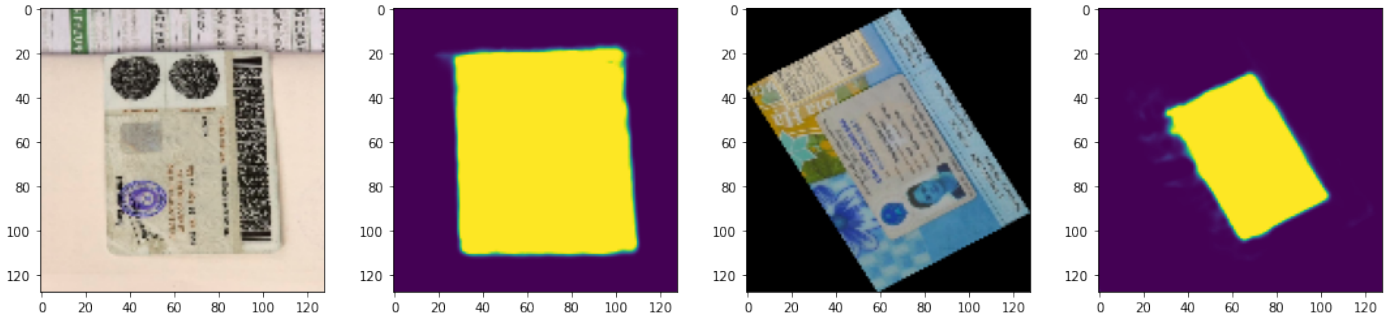
We used U-Net to segment the cards, as shown in Fig. 10. The figure presents an example of using the U-Net model to detect the binary mask of a 12-digit card. We then cropped the card from background images and vertically aligned it, as presented in Fig. 12.



(a) U-Net model

(b) VGG16 model

Fig. 9. Progress of Loss and Accuracy for Training and Testing U-Net and VGG16.



(a) Raw image (back of 12-dgit)

(b) Binary image

(c) Raw image (front of 12-digit)

(d) Binary image

Fig. 10. Segmentation Step (a, b: the Back of a 12-Digit Card and Its Binary Image; c and the Front of a 12-Digit Card and Its Binary Image.)

Experimental results have shown that the proposed methods outperformed similar works in extracting information from the Vietnamese ID card. At the pre-processing step, our method (combination of U-Net, and traditional machine learning) provides more stable results than the work presented in [5], [34]. These methods usually work well in controlled environments, e.g., enough light, clear cards [5], or existence of four corners of cards [34]. For unstable environments, which are very common in many practice applications, they failed in pre-processing the input images, e.g., ID card detection and classification. For this task, thanks to the U-Net model and a rich dataset, our method can respond to unstable situations with high accuracy of 94.0%.

The type of card is important information, but to our knowledge, no existing works presented the way to identify this information. Based on the VGG16 network, we can classify different types of ID cards with high accuracy of 99.5%. Therefore, our method is capable to deal with input images that have many ID cards. Furthermore, this information was very useful at the post-processing step. Regarding text recognition, we obtained a higher accuracy than recent works, such as Hoai *et al.* [6] (98.3% compared with 89.7%) and Viet *et al.* [35] (98.3% compared with 91%).

Besides, owing to the proposed multi-layer architecture, the end-to-end method took an average of 2.5 seconds to extract information from a raw image, as illustrated in Fig. 13. The figure also shows a typical example of the input image that contained similar cards, such as student, business, or license

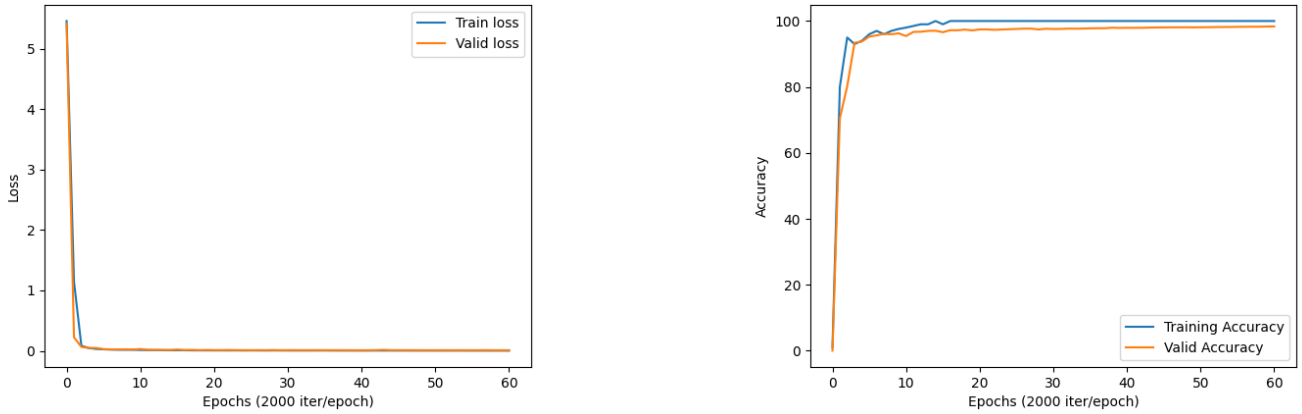


Fig. 11. Progress of Loss and Accuracy for Training and Testing Rebia Network.

(a)

(b)

Fig. 12. An Old ID Card before (a) and after (b) the Pre-processing Step.

cards. With the help of the trained models and algorithms, our method can extract important information accurately.

VI. Conclusion

In this paper, we have presented an end-to-end method to extract information from the Vietnamese ID card. The proposed method contains three consecutive steps: Pre-processing, text detection and recognition, and post-processing. Four neural networks were proposed and trained, which allows us to extract information efficiently, including U-NET for segmentation, VGG16 for classification, CRAFT for text detection, and Rebia for text recognition. We also applied a natural language processing technique (the edit distance) and two image process algorithms (Contour detection and Hough transformation) for layout analysis, text correction, and card alignment.

In addition, a dataset including 3.256 Vietnamese ID cards, 400k manually annotated texts, and more than 500k synthetic texts, was built to validate the proposed methods. We conducted an empirical experiment on our self-collected dataset to demonstrate the effectiveness of the proposed method, which achieved a high accuracy of 94%, 99.5%, and 98.3% for segmentation, classification, and text recognition. These results indicate the promise of the proposed method in the information extraction of similar semi-structured documents. In the future, we will focus on the chip-based ID card and improve the performance of our model, especially the text detector.

References

[1] Y. Zhu, C. Yao, and X. Bai, "Scene text detection and recognition: Recent advances and future trends," *Front. Comput. Sci.*, vol. 10, p. 19–36, Feb. 2016.

[2] N. Nguyen, T. Nguyen, V. Tran, T. Tran, T. Ngo, T. Nguyen, and M. Hoai, "Dictionary-guided scene text recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.

[3] A. Nosseir and O. Adel, "Automatic extraction of arabic number from egyptian id cards," in *Proceedings of the 7th International Conference on Software and Information Engineering*, ICSIE '18, (New York, NY, USA), p. 56–61, Association for Computing Machinery, 2018.

[4] F. M. Rusli, K. A. Adhiguna, and H. Irawan, "Indonesian ID card extractor using optical character recognition and natural language post-processing," *CoRR*, vol. abs/2101.05214, 2021.

[5] T. N. T. Thanh and K. N. Trong, "A method for segmentation of vietnamese identification card text fields," *International Journal of Advanced Computer Science and Applications*, vol. 10, no. 10, 2019.

[6] D. Hoai, H.-T. Duong, and V. Truong Hoang, "Text recognition for vietnamese identity card based on deep features network," *International Journal on Document Analysis and Recognition (IJDAR)*, vol. 24, 06 2021.

[7] R. Soni, B. Kumar, and S. Chand, "Text detection and localization in natural scene images based on text awareness score," *Applied Intelligence*, vol. 49, pp. 1376–1405, 2018.

[8] A. Hazra, P. Choudhary, S. Inunganbi, and M. Adhikari, "Bangla-meitei mayek scripts handwritten character recognition using convolutional neural network," *Applied Intelligence*, vol. 51, pp. 2291–2311, 2021.

[9] X. Ma, K. He, D. Zhang, and D. Li, "Pieed: Position information enhanced encoder-decoder framework for scene text recognition," *Applied Intelligence*, vol. 51, pp. 6698–6707, 2021.

[10] X. Wang, X. Zhang, S. Lei, and H. Deng, "A method of text detection and recognition from receipt images based on CRAFT and CRNN," *Journal of Physics: Conference Series*, vol. 1518, p. 012053, apr 2020.

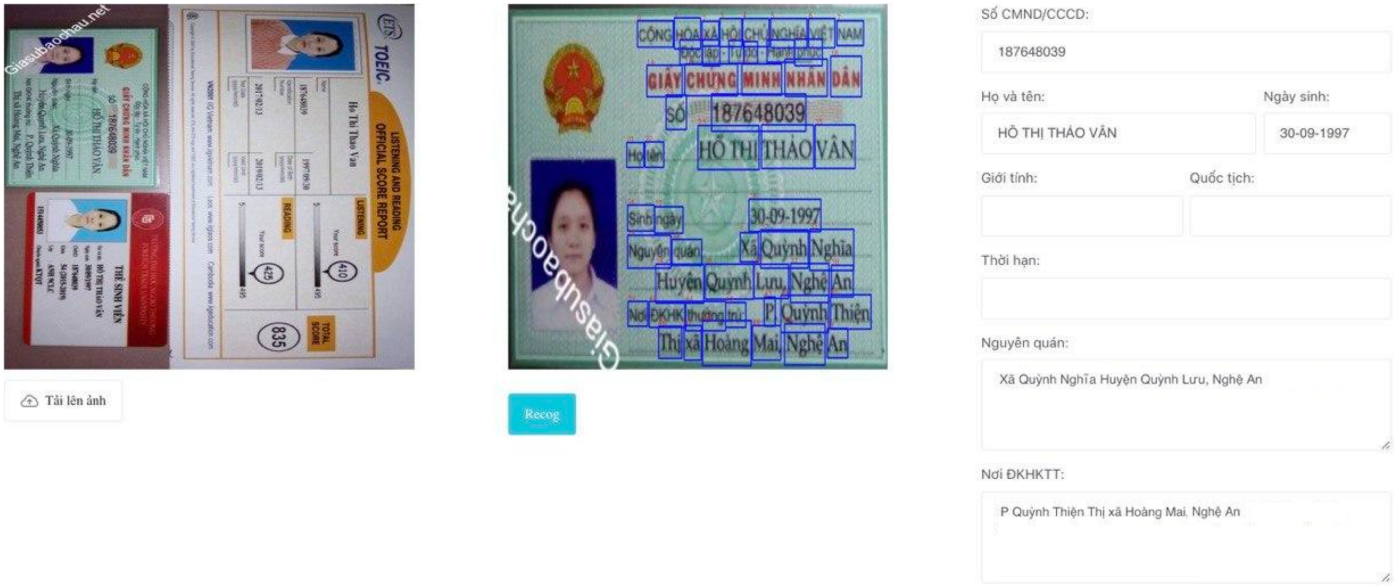


Fig. 13. Extracted Information (the Left Image) from a Raw Image Captured with Several Similar Cards (the Right Image). The Card and Texts were Successfully Segmented, Cropped, Aligned and Detected, as in the Middle Image.

[11] Y. Gao, C. Xu, Z. Shi, and H. Zhang, "Bank card number recognition system based on deep learning," (New York, NY, USA), Association for Computing Machinery, 2019.

[12] M.-C. Ko and Z.-H. Lin, "Cardbot: A chatbot for business card management," in *Proceedings of the 23rd International Conference on Intelligent User Interfaces Companion*, IUI '18 Companion, (New York, NY, USA), Association for Computing Machinery, 2018.

[13] S. Srivastava, S. Sahay, D. Mehrotra, and V. Deep, "Automation of business cards," in *Advances in Interdisciplinary Engineering* (M. Kumar, R. K. Pandey, and V. Kumar, eds.), (Singapore), pp. 371–380, Springer Singapore, 2019.

[14] H. T. Ha, M. Medved, Z. Neverilová, and A. Horak, "Recognition of ocr invoice metadata block types," in *TSD*, 2018.

[15] P. Kumar and S. Revathy, "An automated invoice handling method using ocr," in *Data Intelligence and Cognitive Informatics* (I. Jeena Jacob, S. Kolandapalayam Shanmugam, S. Piramuthu, and P. Falkowski-Gilski, eds.), (Singapore), pp. 243–254, Springer Singapore, 2021.

[16] M. Ryan and N. Hanafiah, "An Examination of Character Recognition on ID card using Template Matching Approach," *Procedia Computer Science*, vol. 59, no. Iccsci, pp. 520–529, 2015.

[17] R. Valiente, M. T. Sadaike, J. C. Gutiérrez, D. F. Soriano, and G. Bressan, "A process for text recognition of generic identification documents over cloud computing," *IPCV'1International Conference on Image Processing, Computer Vision, and Pattern Recognition*, no. April 2017, p. 4, 2016.

[18] A. Alnefaie, D. Gupta, M. H. Bhuyan, I. Razzak, P. Gupta, and M. Prasad, "End-to-end analysis for text detection and recognition in natural scene images," in *2020 International Joint Conference on Neural Networks (IJCNN)*, pp. 1–8, 2020.

[19] Q. Ye and D. Doermann, "Text detection and recognition in imagery: A survey," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 37, no. 7, pp. 1480–1500, 2015.

[20] K. Karthick, K. B. Ravindrakumar, R. Manikandan, and R. Cristin, "Consumer service number recognition using template matching algorithm for improvements in OCR based energy consumption billing," *ICIC Express Letters, Part B: Applications*, vol. 10, no. 10, pp. 895–901, 2019.

[21] Z. Zeng, W. Xie, Y. Zhang, and Y. Lu, "Ric-unet: An improved neural network based on unet for nuclei segmentation in histology images," *IEEE Access*, vol. 7, pp. 21420–21428, 2019.

[22] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2015.

[23] Y. Baek, B. Lee, D. Han, S. Yun, and H. Lee, "Character region awareness for text detection," *CoRR*, vol. abs/1904.01941, 2019.

[24] M. Liao, Z. Wan, C. Yao, K. Chen, and X. Bai, "Real-time scene text detection with differentiable binarization," *CoRR*, vol. abs/1911.08947, 2019.

[25] M. Jaderberg, K. Simonyan, A. Vedaldi, and A. Zisserman, "Synthetic data and artificial neural networks for natural scene text recognition," 2014.

[26] Z. Tian, W. Huang, T. He, P. He, and Y. Qiao, "Detecting text in natural image with connectionist text proposal network," 2016.

[27] J. Baek, G. Kim, J. Lee, S. Park, D. Han, S. Yun, S. J. Oh, and H. Lee, "What is wrong with scene text recognition model comparisons? dataset and model analysis," *CoRR*, vol. abs/1904.01906, 2019.

[28] B. Shi, M. Yang, X. Wang, P. Lyu, C. Yao, and X. Bai, "Aster: An attentional scene text recognizer with flexible rectification," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 41, no. 9, pp. 2035–2048, 2019.

[29] X. Chen, L. Jin, Y. Zhu, C. Luo, and T. Wang, "Text recognition in the wild: A survey," 2020.

[30] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015* (N. Navab, J. Hornegger, W. M. Wells, and A. F. Frangi, eds.), (Cham), pp. 234–241, Springer International Publishing, 2015.

[31] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 248–255, 2009.

[32] T.-H. Pham, X.-K. Pham, and P. Le-Hong, "On the use of machine translation-based approaches for vietnamese diacritic restoration," in *2017 International Conference on Asian Language Processing (IALP)*, pp. 272–275, 2017.

[33] B. Shi, X. Bai, and C. Yao, "An end-to-end trainable neural network for image-based sequence recognition and its application to scene text recognition," 2015.

[34] H. D. Liem, N. Minh, N. B. Trung, H. T. Duc, P. H. Hiep, D. V. Dung, and D. H. Vu, "Fvi: An end-to-end vietnamese identification card detection and recognition in images," *2018 5th NAFOSTED Conference on Information and Computer Science (NICS)*, pp. 338–340, 2018.

[35] H. T. Viet, Q. Hieu Dang, and T. A. Vu, "A robust end-to-end information extraction system for vietnamese identity cards," in *2019 6th NAFOSTED Conference on Information and Computer Science (NICS)*, pp. 483–488, 2019.