INFORMATION EXTRACTION FROM ID CARD VIA COMPUTER VISION TECHNIQUES

Nguyen Thanh Cong, Nguyen Dinh Tuan, Tran Quoc Long

Faculty of Information Technology,
VNU University of Engineering and Technology,
144 Xuan Thuy Street, Cau Giay, Hanoi, Vietnam

Abtract

An identity card is a type of identify documents of each Vietnamese citizen to prove his/her identity. In this modern information technology era, automatically extracting information from identity cards is applied in telecommunications, healthcare, hotels, and more. This report works on the use of computer vision techniques to build a system which extracts information from identity cards. There are three main steps: Identifying identity cards, locating information and extracting information. In particular, identifying indentity cards and locating information are accomplished by basic image processing techniques to increase the speed of the entire system. At the information extraction step, the achievements in text recognition on complex background are used to increase accuracy. After development and experimentation, the system has achieved a relatively high degree of accuracy. The accuracy of the whole system is 95.28%, which makes it possible to be put into practice.

Keywords: ID card, computer vision, convolutional recurrent neural networks, deep learning.

1. Introduction

An identity card is a type of identify documents of each Vietnamese citizen to prove his/her identity. In this modern information technology era, automatically extracting information from identity cards is applied in telecommunications, healthcare, hotels, and more. Nowadays, the data entry is done manually, which takes time and effort.

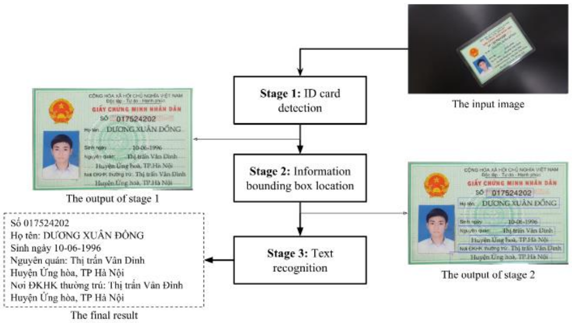
In recent years, computer vision has made a number of achievements in the field of handwriting recognition. One significant example is Tesseract OCRs developed by Hewlett Packard (HP) and funded by Google which are available in 116 different languages and up to 99% accurate. However, the accuracy of Tesseract depends on each of languages which needs recognizing. Tesseract also aims at solving the problem with text documents in the form of scanned images with simple backgrounds. With texts on the complicated background, scientists are still very much interested in and have made certain achievements, but mostly with English language data.

Therefore, this report work on the use of computer vision techniques to build a system which extracts information in Vietnamese language from ID images.

2. Overview of the Approach

Because of the specific features of personal information, this problem requires high and almost absolute accuracy. In order to achieve that requirement, this report proposes some constraints. Firstly, input images are front-side images of identity cards which are put on a solid color background differentiating the ID card background from color. Secondly, there need to be a full picture of the card which enables us to read all details. For example, the identity card is put on white A4 paper. The problem's output is all of extracted information from these cards in a text format.

With such data, the stages of our system are as follows: ID card detection, information bounding box location and text recognition. The system is described visually in figure 1.



Stage 1: ID card detection
Stage 2: Information bounding box location
Stage 3: Text recognition
The input image
The output of stage 1
The output of stage 2
The final result

Số 017524202
Họ tên: DƯƠNG XUÂN ĐÔNG
Sinh ngày 10-06-1996
Nguyên quán: Thị trấn Vân Dinh
Huyện Ứng hòa, TP Hà Nội
Nơi ĐKHK thường trú: Thị trấn Vân Đình
Huyện Ứng hòa, TP Hà Nội

Figure 1: The flow of the system.

Detecting ID card and locating information bounding box are based on image processing techniques to speed up the whole system while maintaining high accuracy for the input image constraints. These two stages act as pre-processing for the text recognition stage.

The text recognition stage shows a text string prediction that occurs in areas detected at the preprocessing stage. Deep learning techniques is applied in this stage to recognize sequence-like objects in an input image. The neural network used is the combination of deep convolutional neural networks (DCNN) and regression neural networks (RNN). It can recognize a sequence which has unknown length such as a sentence or a line of text. It requires only height normalization in both training and testing phases.

3. ID Card Detection Methods

At this stage, ID card is detected and cut out of the background, then standardized in direction and size. With the constraints of the problem, detecting the identity becomes the problem of detecting a rectangle with specific features. The mentioned feature is that the color of the identity card which is completely different from the background. This problem is easier.

In [2], Martin Hirzer proposed a three-step method. Step 1, performing the detection of straight lines that are edges of the rectangle and joining the lines that are likely to belong to the same edge. Step 2, expanding the straight lines in step 1. Finally, the algorithm detects the corners of the rectangle based on the results of step 2.

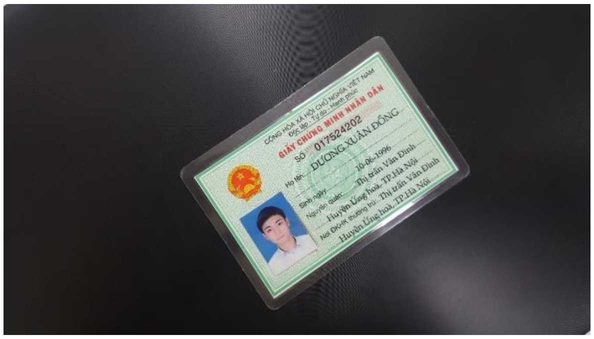


Figure 2: The input of the system.

That is also a way to solve the problem. However, this report proposes a simpler two-step approach: locating identity and standardizing identity. The input of this phase is similar to the figure 2.

The identification of the color identity of the identity card is the identity card in the HSV color model that satisfies:

$$84° \leq H \leq 141°$$

$$8\% \leq S, V \leq 100\%$$

As a result, color filters in the HSV color system are made to binary image. The brightest pixels are the pixels above, the dark pixels are the unsatisfactory. Next, the morphological transformation is used to expand the imagery area and to link the clusters within the identity area. Then select the area with the largest area, this area is the area of ID card in the input image due to the constraints of the problem.

The next step is to find a rectangle with the smallest area containing the remaining light in the binary image. First, the Ramer-Douglas-Peucker algorithm is used to reduce the number of edge of thi area contour, denote by P. Next, consider each edge of the P, with each edge, construct a line containing this edge and denoted by L_1. At the furthest point of P over L_1, construct L_2 parallel to L_1. Next, construct the L_3 line perpendicular to L_1 and L_2, and divide the plane into two halves such that the polygon P is one side of L_3 and L_3 contains at least one of P. Finally, straight line L_4 is parallel to L_3 and pass through the far end of P with respect to L_3. The intersection of the four L_1, L_2, L_3, L_4 lines together is the vertices of the rectangle containing P. So we calculate the area of this rectangle and select the smallest area. The algorithm has a computational complexity of $O(n^2)$, where n is the number of vertices of P. Figure 3 description of the identity container.

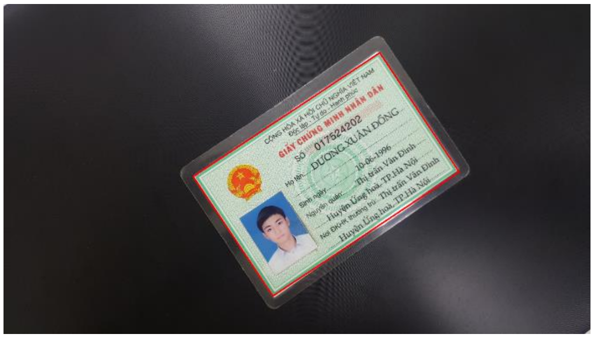


Figure 3: The bounding box of the identity card.

Once you have located the four vertices of the identity container, the next step is to cut and standardize. The need is to standardize both the size and direction of the ID card as it in the input image can be in any size and direction. The direction of a ID card is shown in figure 4. The feature used to determine the direction of the identity card is that the color of the national emblem and the letters of the paper’s name is the same and they completely separate from the background color. The location of these two areas is nearer the top edge of the identity card than the bottom edge. Therefore, color filtering is performed again to retain the following red thresholds:

$$-15° \leq H \leq 15°$$

$$50\% \leq S, V \leq 100\%$$

However, this time only perform color filtering in the image area containing the identity card detected above. Next, erosion morphology transformation is made to reduce the effect of noise. The average distance from the other points to the 2 long sides of the identification is calculated. The top long side of the ID has a smaller distance so we have identified the direction of ID card. As a result, we are able to crop the image and rotate it in the right direction. This report implements the perspective transformation to perform the above two steps simultaneously.

After finishing direction standardization, we continue to standardize the size before moving on to the next step. The actual ID has

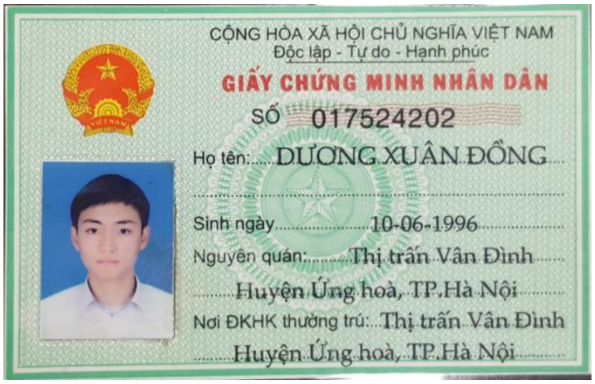

CỘNG HÒA XÃ HỘI CHỦ NGHĨA VIỆT NAM
Độc lập - Tự do - Hạnh phúc
GIẤY CHỨNG MINH NHÂN DÂN
SỐ 017524202
Họ tên:......DƯƠNG XUÂN ĐỒNG......
Sinh ngày......10-06-1996......
Nguyên quán:......Thị trấn Vân Đình......
......Huyện Ứng hoà, TP.Hà Nội......
Nơi ĐKHK thường trú:...Thị trấn Vân Đình
......Huyện Ứng hoà, TP.Hà Nội......

Figure 4: The output of stage 1.

a length-to-width ratio of about 17/11, but in the experiment, the identity detection shown above has not detected the identity correctly. That means that we can mislead the border between the ID card and the background as ID, so the above ratio is not really useful. Thus, the report not only performs the standardization of the height of the identity but also calculates the new width according to the initial proportions of the identity area detected to reduce the distortion of the textual information in the image input. This will increase the accuracy of the information extraction step that will be presented later. During the experiment, the report selects the new heights of the image which is 950 to perform the standardization. With this height, the information in the ID card is relatively easy to see with the human eye and the processing speed of the following steps is not much affected by the size of the image.

4. Text Detection Methods

Locating information areas is the second stage in this system. This stage takes input as the output of the first stage, which is an image just containing 950 x *width* ID card. In this stage, we need to determine rectangles containing the fields of information in ID card, these rectangles are output of second stage.

As we know, ID card has same special properties such as having same font type and background. Based on these properties, we propose a method for locating information areas using image processing techniques. Details, background of ID card is almost green and information content is almost black, so we use color filtering method again. We need to retain information pixel and eliminate not important pixel like background.

In RGB color model, background has high value on G and B channel, in contrast, information part has low value on R, G and B. Based on these characteristics, we use G and B channel to construct grayscale image. And then, we apply equalizing histogram for this grayscale image and choose threshold value to keep information pixel. After apply binary image using above threshold value we will get result consisting components not connected, so we perform dilation morphology for connecting these components.

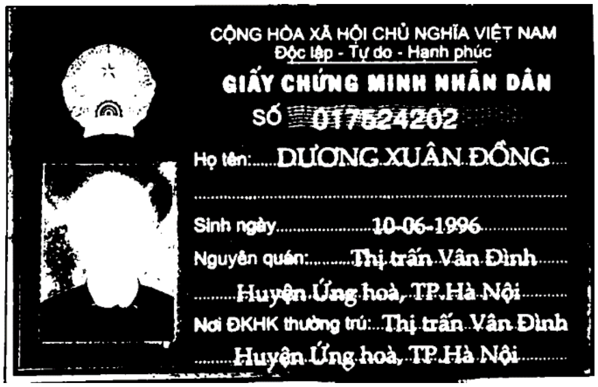
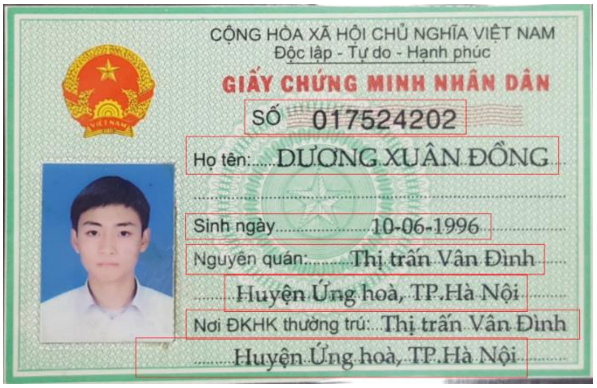

CỘNG HÒA XÃ HỘI CHỦ NGHĨA VIỆT NAM
Độc lập - Tự do - Hạnh phúc
GIẤY CHỨNG MINH NHÂN DÂN
SỐ 017524202
Họ tên:......DƯƠNG XUÂN ĐỒNG......
Sinh ngày......10-06-1996......
Nguyên quán:......Thị trấn Vân Đình......
......Huyện Ứng hoà, TP.Hà Nội......
Nơi ĐKHK thường trú:...Thị trấn Vân Đình
......Huyện Ứng hoà, TP.Hà Nội......

Figure 5: Binary image.

We sequentially find contours of components and then determine bounding box surrounding contours. Before accepting this result, we need to normalize these box such as if box is too large, we can divide it based on reasonable ratio.

From template of ID card, we define 5 boxes consisting 5 information fields of ID card. Combining this template and bounding box, we specify what field bounding box is in. After specifying corresponding field for bounding boxes, we concatenate boxes in same line and eliminate boxes has no field.



CỘNG HÒA XÃ HỘI CHỦ NGHĨA VIỆT NAM
Độc lập - Tự do - Hạnh phúc

GIẤY CHỨNG MINH NHÂN DÂN

SỐ 017524202

Họ tên: DƯƠNG XUÂN ĐỒNG

Sinh ngày 10-06-1996

Nguyên quán: Thị trấn Vân Đình
Huyện Ứng hoà, TP.Hà Nội

Nơi ĐKHK thường trú: Thị trấn Vân Đình
Huyện Ứng hoà, TP.Hà Nội

Figure 6: Final result of stage 2.

5. Text Recognition Methods

Text recognition is a problem in the top of interest in the field of machine vision. The traditional approach consists of three steps: text detection, text segmentation, character recognition, and post-processing. That approach has some drawbacks with the things whose background is complicated such as identity cards. Firstly, the characters are not clearly separated, resulting in the low accuracy. The second limitation is the need for a linguistic model for post-processing.

Recently, Jaderberg et al. [8] used convolutional networks to recognize a whole word without breaking the text into the character level. However, this network requires a fixed input size and the number of network output labels is too large (about 90000 labels) while each word is considered as a label. Consequently, that leads to the number of network parameters is too big (about 409M). Models can not predict untrained words.

In this report, a network architecture called Convolutional Recurrent Neural Networks (CRNN) is used to solve the problem. This architecture was studied in Baoguang Shi al. [9] and Pan He al. [2]. CRNN is a combination of CNN, RNN and CTC. The model architecture is shown in figure 7.

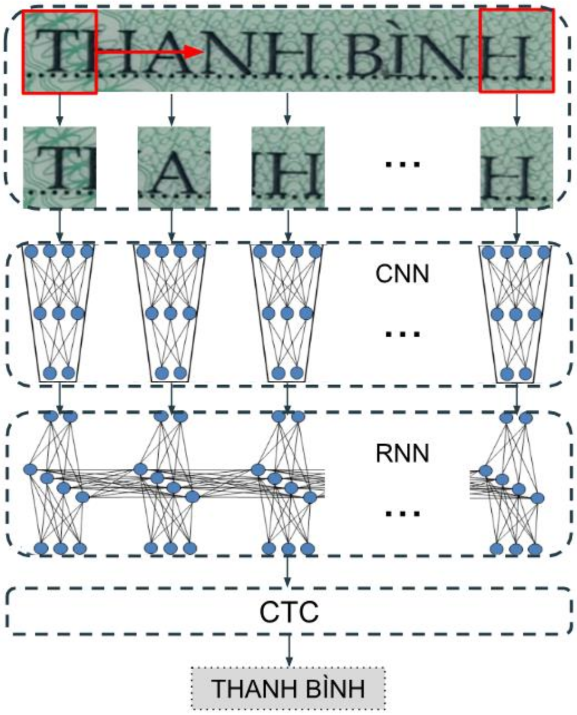


Figure 7: The Convolutional Recurrent Neural Networks Architecture.

From top to bottom, the first thing is the input image, followed by CNN to extract the feature sequence corresponding to each input image. The third is the RNN to label each frame of feature sequence given by CNN. The final thing is the CTC to translate the per-frame predictions by the RNN into the label sequence.

5.1 Feature Extraction

In the CRNN model, the CNN component is essentially a CNN that typically leaves the fully-connected layer at the bottom. With this architecture, CNN is suitable for extracting the features of the input image. Therefore, CNN inputs can be of any size. However, to solve the problem, this report normalizes the height of all input images to 32 pixels and calculates the new width based on the original scale. Thus, the output of the CNN component is a sequence of feature vectors of each frame on the input image. In other words, each feature vector in the output of CNN is features of a local area on the input image.

5.2 Sequence Labeling

After extracting features, the next task is to build the RNN component to label this sequence. According to Baoguang Shi al. [9], RNN have three advantages. Firstly, RNN are able to learn context information of input sequence. Using contextual information with this problem is more stable and useful than recognizing individual characters. For example, some blurry characters in the image can be recognized by the neighboring characters. Secondly, RNN can spread opposite the derivative of the loss function to the input. This allows CNN to be trained with RNN by single loss funtion. And finally, RN can handle sequences of arbitrary length.

However, traditional RNN is unlikely to remember long-term contexts but only short-term ones. Thus, LSTM are designed to overcome this disadvantage. Nevertheless, the LSTM is a directional unit, which uses only information about context in the past. For text image recognition problems, bidirectional context information can be used and complementary to each other. Accordingly, according to Baoguang Shi al. [9], they have used a bidirectional LSTM unit consisting of two LSTM units, one using forward context information and one using backwards information to complement one another. According to Pan He al. [2], deep bidirectional LSTM or just one bidirectional LSTM layer also give similar results. Therefore, the RNN component constructed in this report only includes one bidirectional LSTM layer.

5.3 Transcription

Finally, the output of the RNN is a sequence of probability distributions of each feature vector, which corresponding to a region in the input image. This is not the final output of the network, as large characters in the image need to be identifiable with more specific vectors. Thus, a decoding layer is used at the end of the model. It is a CTC layer. During training, CTC not only is loss funtion but also to decode the output sequence of RNN. During the application process to predict, CTC functions to decode the output chain of RNN.

The network configuration we use in our experiments is summarized in Table 1.

Table 1: Network configuration summary

Layer	Configurations
Input	32 x Width x 3
Conv1	filter=64, size=(3x3), s=1, p=1
MaxPooling1	size=(2x2), s=2
Conv2	filter=128, size=(3x3), s=1, p=1
MaxPooling2	size=(2x2), s=2
Conv3	filter=256, size=(3x3), s=1, p=1
Conv4	filter=256, size=(3x3), s=1, p=1
MaxPooling3	size=(2x2), s=2
Conv5	filter=512, size=(3x3), s=1, p=1
Batch_normalization1	-
Conv6	filter=512, size=(3x3), s=1, p=1
Batch-normalization2	-
MaxPooling4	size=(2x1), s=(2,1)
Conv7	filter=512, size=(2x2), s=1, p=0
Reshape	-
Bidirectional LSTM	hidden units=512
CTC	-

5.4 Network Training

Once the model has been selected, the training process is performed. Since identity card data is personal and no publicly available identification database is available, model training is performed on dummy data and a small amount of real data.

For real data, the data is crop out from the real identity image. For each identification photo, identify all meaningful strings in the valuable information areas of the identity card. For each of those strings, we cut the image region containing only that string into the background in the bounding box of the string without any other characters. Then label this sample. For example, Nguyen can be cut into Ngu, Nguyen, gu, guy, uy, Nguyen, yen. Finally, this report labeled real data on 7 identity cards and collected 1015 training data samples. Figure 8 shows some training data samples and their labels.

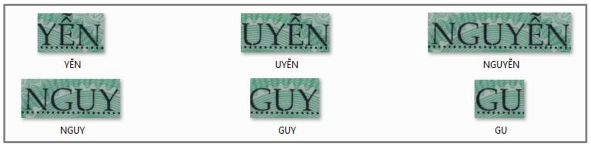


Figure 8: Some real data and their labels.

For dummy data, the report performs sampling of some areas on the empty identity card. Then text data is collected from the administrative units of Vietnam from the commune level upwards. Text data is processed, resulting in 172669 text samples. Then generate the data during the training by writing the text samples onto the cut out areas of the identity card with different fonts. Figure 9 shows some dummy data.

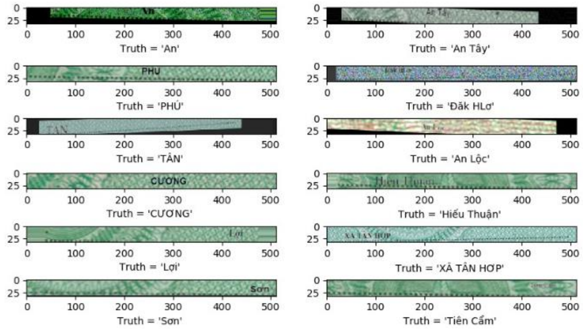


Figure 9: Some fake data.

The network model is trained on the Nvidia GTX 1050 GPU. The number of loops is 10000. Each epoch, the network trained by 6400 data. The average training time is 120s/epoch.

6. Experiments

Data set is obtained from 15 different people in different light condition and angle shooting, then final we have 22 low image and 73 medium image. Below tables describe accuracy of each stage.

Table 2: Accuracy of ID card detection

Data set	Total images	No. images detected correctly	Accuracy (%)
Low image data	22	22	100
Medium image data	73	73	100

Table 3:Accuracy of locating information areas

Data set	Total images	No. images located correctly	Accuracy (%)
Low image data	152	149	98,02
Medium image data	523	522	99,81

Table 4: Accuracy of text recognition

Data set	Accuracy(%)
Training process	98,1
Low image data	87,3
Medium image data	97,6
Total images	95,28

7. Conclusion

The presented approach is real-time OCR for ID card and its accuracy is approximate 95,28%. We don't need to apply any post-processing language model or detect individual character. We collect more than 15 ID cards of different people in different light conditions and shooting angles. In final data set, we have total 95 images consisting 22 low quality and 73 medium quality images.

Specifically, if input image have high quality our system can ensure 97,6% accuracy. In the future, we can collect more data and apply to new ID card types or passport and more.

Acknowledgments

We would like to thank Nguyen Tuan Phong for his fruitful support and discussion while carrying out this research.

References

[1] Alex Graves, Santiago Fernández, Faustino Gomez, Jürgen Schmidhuber, *"Connectionist Temporal Classification: Labelling Unsegmented Sequence Data with Recurrent Neural Networks"*, ICML 2006.

[2] Pan He, Weilin Huang, Yu Qiao, Chen Change Loy, and Xiaoou Tang, *"Reading Scene Text in Deep Convolutional Sequences"*, AAAI 2016.

[3] Martin Hirzer, *"Marker Detection for Augmented Reality Applications"*, Technical report of Graz, 2008.

[4] Sepp Hochreiter and Jürgen Schmidhuber, *"Long Short-Term Memory"*, Neural Computation Volume 9, 1997.

[5] Kyuyeon Hwang, Wonyong Sung, *"Sequence to Sequence Training of CTC-RNNs with Partial Windowing"*, ICML 2016.

[6] Sergey Ioffe, Christian Szegedy, *"Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift"*, ICML 2015.

[7] Max Jaderberg, Karen Simonyan, Andrea Vedaldi and Andrew Zisserman, *"DEEP STRUCTURED OUTPUT LEARNING FOR UNCONSTRAINED TEXT RECOGNITION"*, ICLR 2015.

[8] Max Jaderberg, Karen Simonyan, Andrea Vedaldi and Andrew Zisserman, *"Reading Text in the Wild with Convolutional Neural Networks"*, International Journal of Computer Vision, 2016.

[9] Baoguang Shi, Xiang Bai and Cong Yao, *"An End-to-End Trainable Neural Network for Image-based Sequence Recognition and Its Application to Scene Text Recognition"*, IEEE 2017.