https://doi.org/10.5392/IJoC.2017.13.3.038

Language Identification in Handwritten Words Using a Convolutional Neural Network

Trieu Son Tung and Gueesang Lee
Dept. of ECE,
Chonnam National University, Kwangju, 500-070, Korea

ABSTRACT

Documents of the last few decades typically include more than one kind of language, so linguistic classification of each word is essential, especially in terms of English and Korean in handwritten documents. Traditional methods mostly use conventional features of structural or stroke features, but sometimes they fail to identify many characteristics of words because of complexity introduced by handwriting. Therefore, traditional methods lead to a considerably more-complicated task and naturally lead to possibly poor results. In this study, convolutional neural network (CNN) is used for classification of English and Korean handwritten words in text documents. Experimental results reveal that the proposed method works effectively compared to previous methods.

Key words*: Conventional Neural Network, Korean Text, English Text, Handwritten Document, Classification, Document Analysis.*

1. INTRODUCTION

Over the last few decades, OCR (Optical character recognition) systems have been developed and used for many commercial applications. But in handwriting documents, the handwriting styles differ by individuals and the complexity of classification becomes costly. Accordingly, the difference and the complexity regarding the recognition of the words are due to the writing style, shape, and size of the words.

There has been extensive effort for the design of effective handwritten-word recognition systems, which can be applied to variety of applications such as banking processes, the reading of postal codes, and the others. Because of the wide use of English, a large quantity of Korean documents comprise not only Korean but also English words. Therefore, it became important to further analyze the word-classification process regarding Korean and English.

Typical text recognition system consists of three processing steps: word segmentation, feature extraction, and classification. Word segmentation is a process whereby a string of written language is separated into its component words, and it is for the detection of the interest blocks in the document image. For English and many Latin alpha-based languages, a sound approximation of a word separation is achieved through the use of the inter-word spaces; however, for some languages such as Chinese, Japanese, and Korean, the use of only the spaces between the words is not enough to estimate and segment the words.

* *Corresponding author, Email: gslee@jnu.ac.kr*
Manuscript received Jul. 10, 2017; revised Aug. 08, 2017; accepted Aug. 23, 2017

For the feature-extraction stage, the most-important step of the recognition process, the main goal is the capture of the important symbol characteristics, such as structural features, oriental features, geometrical features and feature extractions based on Hidden Markov Model and Gaussian Mixture Model [1]-[5]. In the last stage of classification, a classifier is used to decide the language type in the word block based on its descriptor. In the works of Zheng et al. [6], [7], the feature extraction is performed by acknowledging the difference between the appearances or the physical structures of the text and the noise. This feature-extraction technique is based on the overlap area between the characters, the run-length histogram, the denseness of the black pixels, and the width, height, and area in each word block. Kuhnke et al. [8] utilized the structural features of the characters, where straight lines that follow both the horizontal and vertical orientations and the symmetry between two sides are the key features. Also, [9] proposed a classification method for which several structural word-block features including deviation of width, height, area, density of the black pixels are used. [10] proposed a method based on the bags-of-word model for which they collected the SIFT features and [11] proposed an automatic script-identification process for the document images for which cluster-based templates are used; here, the extracted feature is the texture of the text. [12] used the text orientation, spacing, capitalization, and interpolations as the basis for the extraction of the textual feature. While these techniques achieved satisfactory results, the performances of these techniques depend on the results of the word-segmentation steps, especially the binarization technique, and the word blocks that are used for the feature extraction must be binarized.

The last classification stage of language identification decides the text type in the word block based on its descriptor,

for which typically the employment of the support vector machine (SVM) or the neural network are involved [6], [7].

In this paper, the convolutional neural network (CNN) is used for recognizing language type in handwritten documents. CNNs are variants of multi-layer perceptrons (MLP) containing one or more convolutional layers and max pooling layers. A convolutional layer consists of a set of weights to process a portion of the input signal. The max pooling layer generates a lower resolution version of convolutional filter outputs by computing the maximum value of filter activations within a specified window[13, 14]. Recently, CNNs have shown promising results for various computer vision or pattern recognition problems, including language identification [15]. However in deep neural network based system of [15], language identification has been studied in conjunction with speech recognition. This paper seeks to find solution of language identification as an individual platform.

The rest of this paper is divided into four sections: Section 2 presents the details of the CNN; next, the proposed method is described in Section 3; the experiment results are given in Section 4; and lastly, Section 5 presents the conclusion.

2. CONVOLUTIONAL NEURAL NETWORK

The CNN consists of a sequence of layers, and each CNN layer transforms one activation volume to another through a differentiable function. As shown in Fig. 1, five main types of layer are applied for the creation of the CNN architecture, as follows: convolutional layer, pooling layer, rectified linear-unit layer (ReLU), fully-connected layer, and loss layer.

Convolutional and pooling layers are alternately included in the low- and middle-level layers. The convolutional layers are included in the odd-numbered layers that consist of the bottom layer, and the even-numbered layers include the pooling layers. The nodes on the convolutional and pooling layers are grouped into feature maps. Each map is connected to one or more of the maps of the previous layer. Each node on a map is connected to a small region of each of the connected maps in the previous layer. The node of the convolution layer extracts features from the input image through a convolution operation on the input nodes, and the node of the pooling layer extracts the features by selecting the maximum value among the input nodes.

The convolutional layer is the core block of a CNN. The convolutional-layer parameters include a set of learnable filters. Each filter is spatially small, but they each extend through the full depth of the input volume; for instance, 5 x 5 x 3 is a typical filter on the first layer. During the forward pass, each filter is slid across the width and the height of the input volume, and the dot products between the entries of the filter and the input can be computed at any position; consequently, a two-dimensional (2D) activation map that provides the responses of that filter at every spatial position is produced. Supposedly, the network will learn the filters that become activated when they see some type of visual feature such as the edge of some orientation or the blotch of some color on the first layer; or eventually, when entire honeycomb or wheel-like patterns are seen on the higher network layers. Each of the filters in each convolutional layer will produce a separate 2D activation map, and these activation maps will be stacked along the depth dimension and will produce the output volume.

The convolutional layer extracts the features through the convolution operation that is either on the input image or the feature maps of the previous layer. Each output map is connected to one or more input maps, while each output node is connected to the input nodes in a small window.

Let $X^n_{(p,i,j)}$ be the activation of a node at point (i, j) on the p^{th} map of the n^{th} layer, and let C^n_p be the set of input maps that is connected to the p^{th} map of the n^{th} layer. Because all of the nodes on a map use the same set of weights, $w^n_{(p,q,u,v)}$ is the weight of the connection from $X^{n-1}_{(p,iS_n+u,jS_n+v)}$ to $X^n_{(p,i,j)}$, where $0 \le u, v \le N_n\text{-}1$, for which N_n is the width and the height of the convolution window that connects layer n-1 and layer n. Equation 1 represents the output of each node, where θ^n_p is a bias and f is an activation function, as follows:

$$X^n_{(p,i,j)}=f(\sum_{q\square c^n_p}\sum_{0\le u,v\le N_n\text{-}1} w^n_{(p,q,u,v)}X^{n-1}_{(p,iS_n+u,jS_n+v)}+\theta^n_p), \quad (1)$$

From Eq. (1), the nodes on a map compute the same feature that is extracted from different locations. When the stride is set to one, the features from all of the possible coordinates are extracted by the convolutional layer; therefore, the important features are not missed even if the feature positions are shifted. Further, in terms of Eq. (1), the convolution of the multiple-input feature maps enables the convolution layers to extract the higher-level features from multiple lower-level features. The post-convolution output-feature map, the size of which is based on the size of the input-feature map, is shown in Eq. (2) and Eq. (3), as follows:

$$\text{width}^n = [(\text{width}^{n-1}\text{-}N_n+1)/S_n] , \quad (2)$$

$$\text{height}^n = [(\text{height}^{n-1}\text{-}N_n+1)/S_n] . \quad (3)$$

The pooling layer, another important part of the CNN, is a kind of non-linear down-sampling. Max pooling, the non-linear function that is most-commonly implemented, separates the input image into a set of non-overlapping rectangles, and for each sub-region, the maximum value is outputted. The output-feature maps engage in a one-to-one correspondence with the input-feature maps. A node at (i,j) is connected to the input nodes in an N x N window with an upper-left corner that is located at (iS, jS). Each node selects the maximum value along the input nodes, as indicated by Eq. (4), which does not require any weight, as follows:

$$X^n_{(p,i,j)}=f(\max_{0\le u,v\le N_n\text{-}1}X^{n-1}_{(p,iS_n+u,jS_n+v)}) . \quad (4)$$

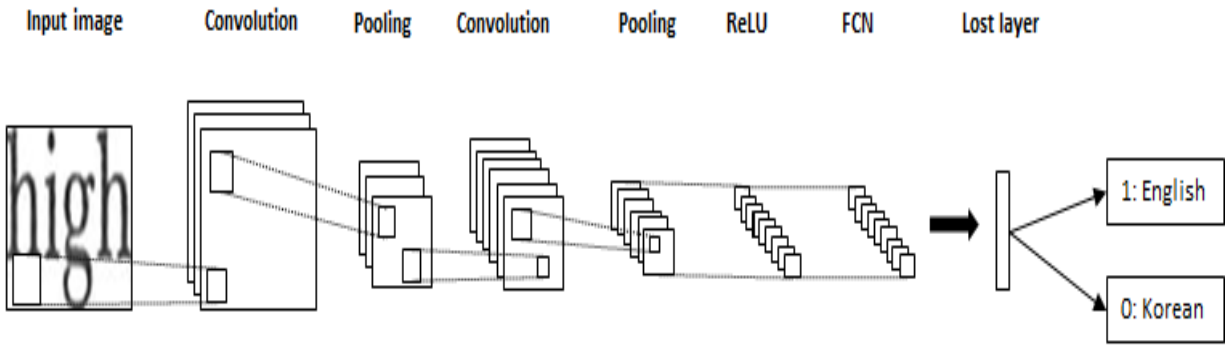

Fig. 1. Convolutional neural network

An alternative to max pooling is average pooling; however, previous studies stated that the results that are generated from max pooling are better than those of average pooling. Similar to the convolution-layer case, the resolution of the output-feature map is decided by Eq. (2) and Eq. (3). The stride of the pooling layer is often set to 2, and because of this, the size of the pooling layer of the feature map is decreased to approximately one-quarter. This layer absorbs the shape variation or distortion, which is an important role in the system. In terms of handwritten characters, the positions of the salient features often shift. The max-pooling node outputs only the maximum value in a window and ignores the offset value; therefore, the max-pooling node still obtains the feature, but the small displacements within the window are not considered. Suppose that the CNN comprises a set of max-pooling layers where each layer absorbs small positional shifts, the network does not require a separate shape normalization to adjust the shape variation. Moreover, the max-pooling layers in a CNN absorb the shape variation in phases, which is suitable for a minimization of the information loss.

The ReLU applies the non-saturating activation function $f(x) = \max(0,x)$. By using this function, the nonlinear properties of the decision function and the overall network are increased while the receptive fields of the convolution layer remain unaffected.

After several of the convolutional and max-pooling layers, the fully-connected layer achieves complete connections with all of the activations in the previous layer. These activations can be calculated with a matrix multiplication that is followed by a bias offset.

The loss layer is the last layer of the CNN architecture. This layer estimates the way that the network training penalizes the deviation between the predicted label and the true label. Various loss functions are used for different tasks; for example, the softmax-loss function is for the prediction of a single class of the K mutually-exclusive classes. To predict the K-independent probability values in [0, 1] and to regress to the real-value labels [-∞, ∞], the Sigmoid cross-entropy-loss function and the Euclidean-loss function are used, respectively.

3. THE PROPOSED METHOD

3.1. Preprocessing step

The proposed English and Korean identification system includes the following two stages: word segmentation and language classification. Word segmentation is performed through connection-component (CC) grouping, and classification is performed by the CNN. There have been numerous method for the binarization [16-21], among which Otsu's method is the most-successful global-thresholding techinique. Once the connected components are identified, noises are eliminated by using bounding box height, width, the density of connected components. The density of a connected component is defined by the ratio of the number of foreground pixels to the total pixel number in the bounding box. The values of the various parameters have been decided so that the text-containing CCs are preserved [9].

To localize each word, the CCs in the same text line and with a distance that is less than half of the average width and height of the CCs, or that comprise overlapping pixels, are united for the formation of the words. The average width is calculated by Eq. (5) and (6), as follows:

$$AW = \frac{\sum_{i=0}^{k} W_i}{2k}, \quad (5)$$

$$AH = \frac{\sum_{i=0}^{k} H_i}{2k}, \quad (6)$$

where i is the number of CCs, and W_i and H_i are the width and the height, respectively, of all of the CCs in the image. The result is illustrated in Fig. 2.

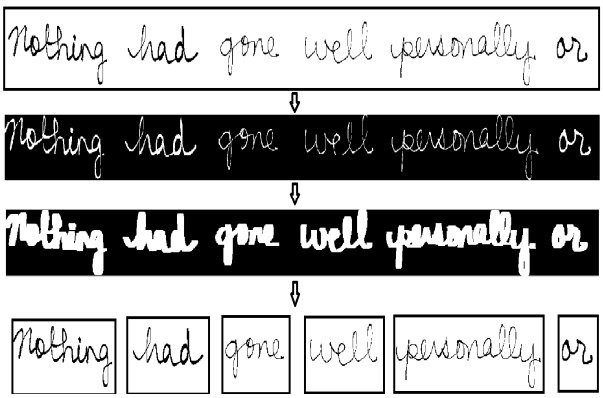

Fig. 2. Word-segmentatio procedure. From top to bottom: original image, binarized image, scanning for the same block word, and word-segmentation result.

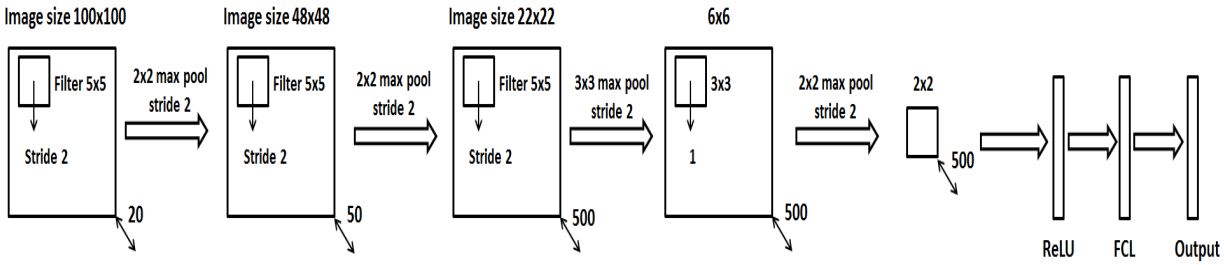

Fig. 3: The proposed method

3.2. Training and classifying with the CNN

For this stage, the CNN is used to classify the Korean and English handwritten word blocks. Three 5 x 5 convolution filters with stride 2, one 3 x 3 convolution filter with stride 1, two 2 x 2 max-pooling filters with stride 2, one 3 x 3 max-pooling filter with stride 2, one ReLU, and one fully-connected layer are used in this network, as is shown in Fig. 3. The learning rate, number of epochs, and batch size are 0.001, 20, and 10, respectively.

4. EXPERIMENT RESULTS

For this project, 260 handwritten-text images from both the English and Korean languages were collected. The project was implemented using Matlab 2015 on the Intel Core i7-4790 CPU at 3.6 GHz with a 4 GB RAM and the Window 7 system. Together with the CUDA 5.5, the GPU that was used for the experiments is the NVIDIA Geforce GT 730. Matcovnet, which is a MATLAB toolbox, was used to create the CNN and for the performance of the training and testing steps. To evaluate the identification performance, 5000 English and Korean handwritten words were used for the training, and 2000 words from each language were used for the testing. All the documents used in the experiment contain English and Korean words together and the words are separated for the language identification. The purpose of the experiments is a comparison between a method for which the bags-of-word model is used and the proposed method. The parameter accuracy that was used for the evaluation of the result was calculated by dividing the number of accurately-classified word blocks by the total number of word blocks and according to the precision and the recall.

Table 1: Experiment results from the bags-of-word (BOW)-based method and the proposed method

	BOW model	Proposed method
Accuracy (%)	79.50	97.56
Precision	86.11	98.16
Recall	79.21	97.83

5. CONCLUSION

This paper presents a classification system for which the CNN is used, wherein the proposed method shows a more-effective result. The training procedure, however, is very intensive in terms of its consumption of time and the computer resources. In the future work, a systematic enhancement solution could be formulated to speed up the training time and to decrease the computer-resource cost.

Fig. 4. Results of the English-word classification

Fig. 5. Results of the Korean-word classification

ACKNOWLEDGMENTS

This research was supported by Basic Science Research Program through the National Research Foundation of Korea(NRF) funded by MEST (NRF-2015R1D1A1A01060172) and by Chonnam National University(Grant number: 2016-2884).

REFERENCES

[1] M. Zissman, "Automatic language identification using gaussian mixture and hidden markov models," International Conference on Acoustics, Speech, and Signal Processing (ICASSP), vol. 2, Apr. 1993, pp. 399-402.
[2] K. P. Li, "Automatic language identification using syllabic spectral features," International Conference on Acoustics, Speech, and Signal Processing (ICASSP), vol. 1, Apr. 1994, pp. I/297-I/300.
[3] R. C. F. Tucker, M. Carey, and E. Parris, "Automatic language identification using sub-word models," International Conference on Acoustics, Speech, and Signal Processing (ICASSP), vol. 1, Apr. 1994, pp. I/301-I/30.
[4] W. B. Cavnar and J. M. Trenkle. "N-gram based text categorization," Proceedings of the Third Annual Symposium on Document Analysis and Information Retrieval, 1994, pp. 161-169.
[5] M. J. Martino and R. C. Paulsen, "Natural language determination using partial words," U.S. Patent No. 6216102 B1, 2001.
[6] Y. Zheng, H. Li, and D. Doermann, "Machine Printed Text and Handwriting Identification in Noisy Document Image," ICDAR, Sep. 2003.
[7] Y. Zheng, C. Liu, and X. Ding, "Single Character Type Identification," Proc. SPIE Conf. Document Recognition and Retrieval, 2002, pp. 49-56.
[8] K. Kuhnke, L. Simoncini, and Zs. M. Kov'acs-V, "A system for machine-written and hand-written character distinction," IEEE Proc. ICDAR, vol. 2, 1995, pp. 811-814.
[9] L. F. Silva and A. Sanchez, "Automatic Discrimination between Printed and Handwritten Text in Documents," ICDAR, 1999.
[10] K. Zagoris, I. Pratikakis, A. Antonacopoulos, B. Gatos, and N. Papamarkos, "Distinction between handwritten and machine-printed text based on the bag of visual words model," PR, 2013.
[11] J. Hochberg, P. Kelly, T. Thomas, and L. Kerns, "Automatic Script Identification from Document Images Using Cluster-Based Templates," IEEE Transactions on PAMI, vol. 19, no. 2. Feb. 1997.
[12] A. Lawrence Spitz, "Determination of the Script and Language Content of Document Images," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 19, no. 3, Mar. 1997.
[13] Convolutional Neural Networks (LeNet), "DeepLearning 0.1 documentation," DeepLearning 0.1. LISA Lab, Retrieved 31 Aug. 2013.
[14] M. Matsugu, K. Mori, Y. Mitari, and Y. Kaneda, "Subject independent facial expression recognition with robust face detection using a convolutional neural network," Neural Networks, vol. 16, no. 5, 2003, pp. 555-559.
[15] Y. Lei, L. Ferrer, A. Lawson, M. McLaren, and N. Scheffer, "Application of Convolutional Neural Networks to Language Identification in Noisy Conditions," Speech Technology and Research Laboratory, SRI International.
[16] N. Otsu, "A threshold selection method from gray-level histogram," IEEE Trans. Syst. Man Cybern., vol. 9, issue. 1, 1979, pp. 62-66.
[17] J. Bernsen, "Dynamic thresholding of gray level images," ICPR, 1986, pp. 1251-1255.
[18] G. Johannsen and J. Bille, "A threshold selection method using information measures," ICPR, 1982, pp. 140-143.
[19] N. J. Kapur, P. K. Sahoo, and C. K. A. Wong, "A new method for gray-level picture thresholding using the entropy of the histogram," JCVPIP, vol. 29, issue. 3, 1985, pp. 273-285.
[20] J. Sauvola and M. Pietikainen, "Adaptive document image binarization," Pattern Recognition, vol. 33, issue. 2, 2000, pp. 225-236.
[21] W. Niblack, *An introduction to digital image processing*, Prentice Hall, Eaglewood Cliffs, 1986, pp. 115-116.

Trieu Son Tung
He received the B.S. degree in Microelectronics from Hanoi University of Science and Technology (HUST) in 2014. He is currently an M.S. student at the Electronics and Computer Science department of Chonnam National University, Republic of Korea. His research interests are multimedia and image processing, vision tracking, and pattern recognition.

Gueesang Lee
He received the B.S. degree in Electrical Engineering and the M.S. degree in Computer Engineering from Seoul National University, Republic of Korea, in 1980 and 1982, respectively. He received the Ph.D. degree in Computer Science from Pennsylvania State University in 1991. He is currently a professor of the Department of Electronics and Computer Engineering at Chonnam National University, Republic of Korea. His primary research interests are image processing, computer vision, and video technology.