Korean letter handwritten recognition using deep convolutional neural network on android platform

S Purnamawati1*, D Rachmawati2, G Lumanauw1, R F Rahmat1, R Taqyuddin1

1Department of Information Technology, Faculty of Computer Science and Information Technology, Universitas Sumatera Utara, Medan, Indonesia
2Department of Computer Science, Faculty of Computer Science and Information Technology, Universitas Sumatera Utara, Medan, Indonesia

sarah_purnamawati@usu.ac.id

Abstract. Currently, popularity of Korean culture attracts many people to learn everything about Korea, particularly its language. To acquire Korean Language, every single learner needs to be able to understand Korean non-Latin character. A digital approach needs to be carried out in order to make Korean learning process easier. This study is done by using Deep Convolutional Neural Network (DCNN). DCNN performs the recognition process on the image based on the model that has been trained such as Inception-v3 Model. Subsequently, re-training process using transfer learning technique with the trained and re-trained value of model is carried though in order to develop a new model with a better performance without any specific systemic errors. The testing accuracy of this research results in 86.9%.

Keywords: Word Recognition, Korean Alphabet Recognition, Handwritten Recognition, Deep Convolutional Neural Network, Transfer Learning

1. Introduction

Popularity of Korean culture or Korean wave (hallyu) has become a phenomenon that draws public attention nowadays, especially in Indonesia. Many young Indonesians are motivated to learn Korean language and to learn it, they try to understand Korean non-latin character (Hangul) that consists of vocal and consonant [1]. Bahasa Indonesia uses latin character that makes Indonesian learners have to deal with a set of difficulties in learning Hangul.

The used method in this research is Deep Convolutional Neural Network. This method showed an outstanding performance in image recognition, because DCNN has a good ability to extract high-level features. The paper was organized as follows: Sect. 2 reviewing several previous researches related to Hangul recognition and Deep Convolutional Neural Network. Section 3 describing the used method in recognizing Korean letter handwritten Section 4, explaining the result of this research. Section 5 discussing the summary and suggestions for future research.

2. Related work

There are a lot of researches about Korean character (Hangul) conducted with various determination and method. Kang used Hierarchical Stochastic Modelling for Hangul character recognition [2]. Sari proposed Template Matching and K-Nearest Neighbors method to recognize Hangul with accuracy rate 85% for Template Matching and 72% for K-Nearest Neighbors [3]. Our previous work also in character recognition is in [11]
There was Hangul handwritten recognition done by Nasution. She used Backpropagation Neural Network with accuracy rates for memorization and generalization were 98.2% and 85.3%. The output of this system was very simple because it was still using GNU GCC MinGW Compiler [4].

Prabowo conducted research on utilization API Gesture to practice writing Hangul in android smartphone. However, the result from the system was not processed for any specific purpose [5].

In [6], they used Deep Convolutional Neural Network for recognizing Hangul handwritten with accuracy rate 95.96% for SERI95a dataset and 92.92% for PE92 dataset. The output of this research was desktop application.

Other similar research was also done by Bui and Chang. They used Deep Convolutional Neural Network for recognizing non-MNIST dataset, which was digit handwritten dataset with more noise and hard to recognize. This research achieved best accuracy rate of 98% [7].

3. Methodology
The used method in this research consisted of several steps, namely pre-trained neural network with inception-v3 model as an outcome, retraining process using transfer learning technique, and word recognition process. Word recognition process was done with following steps, image acquisition, pre-processing and recognition using Deep Convolutional Neural Network.

Figure 1 describes the general architecture of the proposed method.

3.1 Pre-trained neural network
Pre-Trained Neural Network phase was used to train ImageNet Large Scale Visual Recognition Challenge (ILSVRC) dataset which produced a deep learning model called Inception-v3 Model. This model had an awesome capability in recognizing image.

DCNN method was used in the training process since it had outstanding performance in image recognition. DCNN’s capability to recognize image was supported by its feature to extract high-level features. In addition, convolutional layer and max pooling layer could recognize various object. DCNN could conduct every phase in image recognition, since DCNN received raw image as an input, thus it did not need feature extraction phase such as conventional classifier [6]. Figure 2. shows a DCNN’s architecture.
DCNN consists of 5 convolutional layers and three fully-connected layers. The output of fully-connected layer is 1000-way-softmax and produces a distribution over 1000 class labels. Training process of each layer is shown in Table 1.

Table 1. Training Process of Each Layer

| Layer               | Kernel | Size                  | Neuron |
|---------------------|--------|-----------------------|--------|
| Input Image         | -      | 224x224x3x1           | 150.528|
| 1<sup>st</sup> Convolutional Layer | 96     | 55x55x48x2            | 290.400|
| 2<sup>nd</sup> Convolutional Layer | 256    | 27x27x128x2           | 186.624|
| 3<sup>rd</sup> Convolutional Layer | 384    | 13x13x192x2          | 64.896 |
| 4<sup>th</sup> Convolutional Layer | 384    | 13x13x192x2          | 64.896 |
| 5<sup>th</sup> Convolutional Layer | 256    | 13x13x128x2         | 43.264 |
| Fully-Connected Layer | -      | 2048x2              | 4096   |
| Fully-Connected Layer | -      | 2048x2              | 4096   |
| Fully-Connected Layer | (SoftMax Output) | - 1000x1       | 1000   |

The input image was filtered by the first Convolutional Layer with 96 kernels of size 11x11x3 with a stride of 4 pixels. The output of the first Convolutional Layer was filtered by the second. Convolutional Layer with 256 kernels of size 5x5x48. The third, fourth and fifth Convolutional layer was connected to each other without intervention pooling and normalization layers. The output of second Convolutional Layer connected to the third Convolutional Layer with 384 kernels of size 3x3x256. The fourth and fifth Convolutional Layer had 256 kernels of size 3x3x192.

3.2 Retraining process (transfer learning)

Inception-v3 Model could not recognize characters and words that written in Hangul, therefore a retraining process using transfer learning technique with a dataset of Hangul handwritten characters needed. In this phase, bottleneck value of each image was calculated. Bottleneck is an informal term aim to calculate values of each image that will be used for classification. Figure 3. shows an example bottleneck values of A-Ju word.
3.3 Output (new model)
The output of retraining process is a new model that able to recognize handwritten of Korean words with Hangul characters.

3.4 Word Recognition
Word recognition has several steps as follows.
- Input: The input image is image that written on screen smartphone android.
- Preprocessing: There are 2 steps in preprocessing phase as follows.
  - Thresholding: Thresholding is a conversion of input image to binary image [9]. Thresholding formula can be described in formula 1.
    \[ g(x, y) = \begin{cases} 
    1 & \text{if } f(x, y) \geq T \\
    0 & \text{if } f(x, y) < T 
    \end{cases} \]  
  - Resizing: The input image was resized to become 224 x 224.
  - Recognition: In this phase, the result of image preprocessing was recognized by using DCNN method.
  - Output: The output of this research was a word that successfully recognized with translation and pronunciation features.

4. Experimental results
The obtained results from the training and application testing were discussed in this section.

4.1 Training results
Dataset in this research consisted of 11 Korean words with 1157 images, which 1100 of them were used as training data and the rest was used as testing data. The dataset of this research shows in Table 2.
Table 2. Dataset of This Research

| No | Words | Pronunciation | Translation |
|----|-------|---------------|-------------|
| 1  | 아주   | A-Ju          | Very        |
| 2  | 아니   | A-Ní          | No / Not    |
| 3  | 안녕   | An-Nyeong     | Hello / Hey |
| 4  | 보여   | Bo-Yeo        | Can You See That |
| 5  | 어디   | Eo-Dí         | Where       |
| 6  | 어서   | Eo-Seo        | Soon        |
| 7  | 고기   | Go-Gí         | Meat        |
| 8  | 입다   | Ib-Da         | Using       |
| 9  | 주다   | Ju-Dá         | Giving      |
| 10 | 마자   | Ma-Ja         | That’s Right |
| 11 | 모라   | Mo-la         | Don’t Know  |

The retrain process was done by using Tensorflow. Tensorflow is an open source library made by Google for machine learning that supports some of programming language [10]. Bottleneck value of each image was calculated first. The training was conducted with 4000 training steps. During training process, the 10 images were chosen randomly from the training set in each step, then bottleneck value was processed into final layer to get prediction value. There are 7 main parameters of this training, namely Training Steps, Learning Rate, Testing Percentage, Validation Percentage, Eval Step Interval, Train Batch Size dan Validation Batch Size. Table 3. shows main parameters in this training.

Table 3. Main Parameters

| PARAMETER                | PARAMETER’S VALUE |
|--------------------------|-------------------|
| Training Steps           | 4000 steps        |
| Learning Rate            | 0.01              |
| Testing Percentage       | 10%               |
| Validation Percentage    | 10%               |
| Eval Step Interval       | 10                |
| Train Batch Size         | 100               |
| Validation Batch Size    | 100               |

Each training step indicated training accuracy, validation accuracy and cross entropy. Cross entropy is a loss function which shows how well the learning process is running. The lower the value of cross entropy, the better the learning process. If the cross entropy keeps going downwards, it means the learning process improved. If the training accuracy is high but the validation accuracy is low, that means the network is overfitting, which the network is memorizing some feature that are not useful for recognizing image. The final accuracy is displayed after the training process. Summary of the training process shows in Table 4.
Table 4. Summary of Training Process

| Training Steps | Training Accuracy (%) | Cross Entropy | Validation Accuracy (%) |
|----------------|------------------------|---------------|-------------------------|
| 0              | 15.0                   | 2.342418      | 11                      |
| 250            | 100                    | 0.336106      | 100                     |
| 500            | 99                     | 0.193195      | 92                      |
| 750            | 99                     | 0.112672      | 96                      |
| 1000           | 100                    | 0.115776      | 99                      |
| 1250           | 100                    | 0.082326      | 100                     |
| 1500           | 100                    | 0.067476      | 100                     |
| 1750           | 100                    | 0.074174      | 98                      |
| 2000           | 100                    | 0.056418      | 97                      |
| 2250           | 100                    | 0.044302      | 97                      |
| 2500           | 100                    | 0.056943      | 99                      |
| 2750           | 100                    | 0.044161      | 100                     |
| 3000           | 100                    | 0.030162      | 98                      |
| 3250           | 100                    | 0.036282      | 100                     |
| 3500           | 100                    | 0.032134      | 100                     |
| 3750           | 100                    | 0.022177      | 100                     |
| 3999           | 100                    | 0.035141      | 96                      |

Final Accuracy: 100%

Figure 4. shows training accuracy graph in this research that has value range accuracy [0,1] which value 0 means minimum of accuracy value and 1 means maximum of accuracy value in 4000 training steps.

Cross entropy graph of this 4000 training steps in this research is shown in Figure 5. In Figure 5, the value of cross entropy kept downwards, means the system had a better performance in recognizing Hangul.
4.2. Testing Results
We tested Korean Letter Handwritten recognition with 57 images. In this application, the example of how to write the words was shown above the canvas. To start the recognition, firstly, the image was inputted by writing a word in a a screen of smartphone android and then the detect button pressed. The pronunciation and translation of the word would be displayed. If the play button pressed, the voice of pronunciation would be played. The recognition process is show in Figure 6.

![Cross Entropy Graph](image)

**Figure 5.** Cross Entropy Graph

![Recognition process](image)

**Figure 6.** Recognition process

We conducted the application testing with 14 people. They wrote different words in the different numbers. The result of testing system is shown in Table 5.
Table 5. The Result of Testing System

| Image | Word  | Accuracy (%) | Image | Word  | Accuracy (%) |
|-------|-------|--------------|-------|-------|--------------|
| 1     | A-Ju  | 97.8         | 29    | Eo-Seo| 94.3         |
| 2     | A-Ju  | 67.5         | 30    | Eo-Seo| 55.4         |
| 3     | A-Ju  | 69.3         | 31    | Eo-Seo| 99.3         |
| 4     | A-Ju  | 90.7         | 32    | Eo-Seo| 94.9         |
| 5     | A-Ni  | 66.5         | 33    | Go-Gi | 98.9         |
| 6     | A-Ni  | 96.2         | 34    | Go-Gi | 63.6         |
| 7     | A-Ni  | 81.1         | 35    | Go-Gi | 68.3         |
| 8     | A-Ni  | 88.5         | 36    | Go-Gi | 95.4         |
| 9     | A-Ni  | 90.4         | 37    | Go-Gi | 95.7         |
| 10    | A-Ni  | 93.5         | 38    | Go-Gi | 94.5         |
| 11    | An-Nyeong| 61.0     | 39    | Ib-Da | 90.6         |
| 12    | An-Nyeong| 90.8     | 40    | Ib-Da | 98.5         |
| 13    | An-Nyeong| 96.4     | 41    | Ib-Da | 72.2         |
| 14    | An-Nyeong| 98.8     | 42    | Ib-Da | 92.9         |
| 15    | An-Nyeong| 99.5     | 43    | Ju-Da | 91.2         |
| 16    | An-Nyeong| 68.8     | 44    | Ju-Da | 72.1         |
| 17    | Bo-Yeo | 97.9         | 45    | Ju-Da | 86.1         |
| 18    | Bo-Yeo | 87.1         | 46    | Ju-Da | 95.9         |
| 19    | Bo-Yeo | 89.9         | 47    | Ju-Da | 87.3         |
| 20    | Bo-Yeo | 90.6         | 48    | Ma-Ja | 62.1         |
| 21    | Bo-Yeo | 70.2         | 49    | Ma-Ja | 98.4         |
| 22    | Eo-Di  | 68.0         | 50    | Ma-Ja | 97.3         |
| 23    | Eo-Di  | 96.1         | 51    | Ma-Ja | 96.6         |
| 24    | Eo-Di  | 95.2         | 52    | Ma-Ja | 93.4         |
| 25    | Eo-Di  | 90.3         | 53    | Ma-Ja | 97.5         |
| 26    | Eo-Seo | 71.4         | 54    | Mo-lIa| 91.7         |
| 27    | Eo-Seo | 96.1         | 55    | Mo-lIa| 93.1         |
| 28    | Eo-Seo | 95.8         | 56    | Mo-lIa| 92.1         |
|       |        |              | 57    | Mo-lIa| 72.7         |

Accuracy Average: 86.9%

Overall, the detection performed with accuracy average achieve of 86.9%. There were several problems affecting accuracy of each image. The first problem was the stroke shape of the words. The stroke shape would be conformed with the image training. The abrupt stroke of the words affected the accuracy value too.

5. Conclusion and Future Work

Based on testing result of Korean letter handwritten recognition using Deep Convolutional Neural Network on Android platform, the average accuracy of 86.9% was achieved. Accuracy value of input image was affected by the stroke shape of Hangul. 100 % of training accuracy was acquired in each word which had 100 images. The number of trained images in each word affected the training accuracy. In the future, the number of word classes are possible to be added. System can recognize more than two syllables and cursive letters.
References

[1] Damarullah W, Hamzah A, and Lestari U 2013 Aplikasi Pengenalan dan Pembelajaran Bahasa Korea (Hangul) Berbasis Android Journal Script vol.1 No.1 pp 78-88

[2] Kang K W and Kim J H 2003 Utilization of Hierarchical, Stochastic Relationship Modelling for Hangul Character Recognition. IEEE Transaction On Pattern Analysis And Machine Intelligence vol.26 no.9 pp 1185-1196

[3] Sari K P, Tritoasmoro I I and Susatio E 2012 Pengalih Citra Karakter Korea-Indonesia Menggunakan Klasifikasi K-Nearest Neighbour Dan Template Matching. Skripsi. Telkom University

[4] Nasution R H 2014 Pengenalan Kata Tulisan Tangan Huruf Korea Menggunakan Jaringan Saraf Tiruan Propagasi Balik. Skripsi. Universitas Sumatera Utara

[5] Prabowo Y D, Supraptono E and Safi’I E 2016 Efektivitas Aplikasi Berbasis Android Menggunakan API Gesture Untuk Mengenal Aksara Korea. Didaktikum: Jurnal Penelitian Tindakan Kelas vol.17 no.2 pp 78-82

[6] In-Jung K. and Xie X 2014 Handwritten Hangul Recognition using Deep Convolutional Neural Networks Int. J. Document Analysis and Recognition (IJDAR) vol.18 pp 1-13

[7] Bui V and Chang L 2016 Deep Learning Architectures for Hard Character Classification Int. Conf. Artificial Intelligence: pp 108-114.

[8] Krizhevsky A, Sutskever I and Hinton G E 2012 ImageNet Classification with Deep Convolutional Neural Networks. Advances in Neural Information Processing Systems (NIPS) pp 1097-1105

[9] Yadav S A, Sharma S and Kumar S R 2015 A Robust Approach For Offline English Character Recognition. 1st Int. Conf. on Futuristic Trend in Computational Analysis and Knowledge Management pp 121-126

[10] Devikar P 2016. Transfer Learning for Image Classification of Various Dog Breeds. Int. J. of Advanced Res. Comput. Eng. Technol. (IJARCET) vol 5 pp 2707-2715

[11] Gunawan D, Arisandi D, Ginting F M, Rahmat R F and Amalia A 2017 Russian character recognition using Self-Organizing Map J. Phys. Conf. Ser. 801, p. 12040