J Inf Process Syst, Vol.16, No.6, pp.1437~1446, December 2020
https://doi.org/10.3745/JIPS.02.0149

ISSN 1976-913X (Print)
ISSN 2092-805X (Electronic)

JOURNAL OF INFORMATION PROCESSING SYSTEMS JIPS

Design of Image Generation System for DCGAN-Based Kids' Book Text

Jaehyeon Cho* and Nammee Moon**

Abstract
For the last few years, smart devices have begun to occupy an essential place in the life of children, by allowing them to access a variety of language activities and books. Various studies are being conducted on using smart devices for education. Our study extracts images and texts from kids' book with smart devices and matches the extracted images and texts to create new images that are not represented in these books. The proposed system will enable the use of smart devices as educational media for children. A deep convolutional generative adversarial network (DCGAN) is used for generating a new image. Three steps are involved in training DCGAN. Firstly, images with 11 titles and 1,164 images on ImageNet are learned. Secondly, Tesseract, an optical character recognition engine, is used to extract images and text from kids' book and classify the text using a morpheme analyzer. Thirdly, the classified word class is matched with the latent vector of the image. The learned DCGAN creates an image associated with the text.

Keywords
DCGAN, NLTK, OCR

1. Introduction

Generally, childhood education institutions, books, or language areas have been developed for young children to have access to a variety of language activities and books. In this area, preschoolers can read the books themselves or teachers can read the books to the children. Kids' book reading activity is frequently used and is considered important by early childhood education institutions [1]. However, in recent years, digital kids' book on tablet PCs and smartphones have begun to replace physical books [2]. Academia has determined that smart devices affect children lives and development.

According to a study, 90.3% and 21.9% of infants under seven years of age have used smartphones and tablet PCs, respectively. In addition, studies have also shown that the use rate of smart devices among children aged 4 to 6 years is approximately 95%. Such studies suggest that smart devices have already become a familiar medium for most children. Currently, research is being conducted on the use of smart devices for children in both the fields of special education and general education. In terms of educational effects, smart devices can increase synesthesia and communication with Internet access, voice, video, text, etc., and increase creativity through app-based language activities. Fig. 1 illustrates a photograph that uses the kids' book and smart device to educate children.

Manuscript received Febryary 12, 2019; first revision Febryary 11, 2020; accepted Febryary 24, 2020..

Corresponding Author: Nammee Moon (mnm@hoseo.edu)
* Division of Computer Engineering, Hoseo University, Korea (jaehyeon99@naver.com)
** Division of Computer and Information Engineering, Hoseo University, Korea (mnm@hoseo.edu)

Fig. 1. A kid is being educated using smart device.

In 2014, Goodfellow et al. [4] proposed a generative adversarial network (GAN), wherein generators replicated texts that were classified such that they became hostile toward each other, thus, promoting learning and enhancement of performance. This differentiated the tendency of machine learning research from supervised learning, such as a convolutional neural network (CNN) and recurrent neural network (RNN), to semi-supervised learning and unsupervised learning [5].

The security of datasets is complex in machine learning. In case of vast data in supervised learning, classifying the dataset is complex and meaningful learning is often difficult owing to noise. Sorting secured data and removal of noise can be time consuming and costly. Conversely, for small data, it is difficult to avoid overfitting.

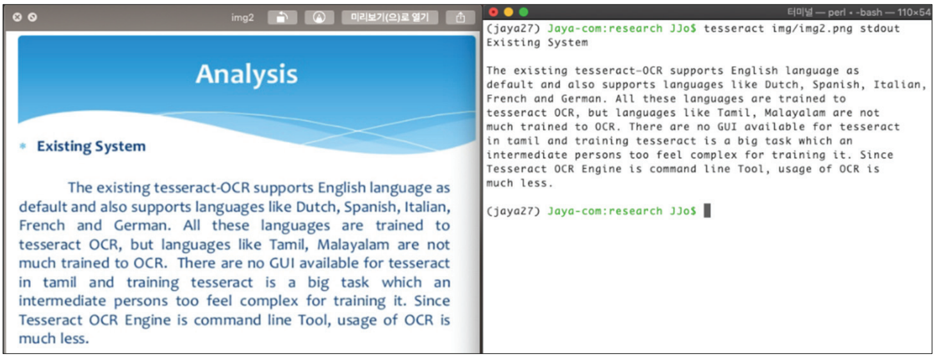
This issue can be addressed with the use of a GAN, which transforms and creates an image through random noise; and the classifier learns its classification probability through a comparison with the original and the replication probability of the generator through the minimax algorithm. A GAN is a model for simultaneous learning by the classifiers and generators, and it solves the problem of generation and classification, while existing machine learning methods solve classification, recognition, and prediction problems. These characteristics are utilized or studied in various fields such as reproducing ink paintings [6] and producing music [7].

This study proposes the extraction of images and texts from kids' book with the use of smart devices and creation of new images for text in kids' book by matching images and text. Such a system will be of significant importance when using smart devices as an educational media for children. For text extraction, the image and text are obtained from the kids' book, and the text is extracted from the captured image using optical character recognition (OCR). The extracted text is classified into nouns and verbs by using a Natural Language Toolkit (NLTK) morpheme analyzer, and it teaches by matching objects, actions, and images using a discriminator. It considers the images of texts from kids' book that have no images and designates the noun and verb of the photographed text as the latent vector of the generator. The learned generator creates a similar image that fits the text.

2. Related Work

2.1 OCR

OCR, also known as optical character reader, is the electronic or mechanical conversion of images of typed, handwritten, or printed text into machine-encoded text, either from a scanned document, a photo of a document, a scene-photo, or subtitle text superimposed on an image. The OCR service allows the conversion of scanned images, faxes, screenshots, portable document format (PDF) documents, and eBooks to text. OCR is designated as the beginning of research in the fields of machine learning and deep learning. Previously, OCR and digital character recognition were considered to be different domains; however, at present, the term optical character recognition includes digital character recognition as well. OCR uses optical techniques such as mirrors and lenses, and digital character recognition performs character recognition by using scanners and algorithms. Although most of the font styles can be converted with high probability, the initial system requires training to identify the font style for text readability. Modern systems can generate document files that closely match text images from the read images. Some of them are properly recognized even if they contain parts that are not present in the document, as shown in Fig. 2 where text is being read from an image using OCR [8,9].



Analysis

- Existing System

The existing tesseract-OCR supports English language as default and also supports languages like Dutch, Spanish, Italian, French and German. All these languages are trained to tesseract OCR, but languages like Tamil, Malayalam are not much trained to OCR. There are no GUI available for tesseract in tamil and training tesseract is a big task which an intermediate persons too feel complex for training it. Since Tesseract OCR Engine is command line Tool, usage of OCR is much less.

```
(jaya27) Jaya-com:research JJo$ tesseract img/img2.png stdout
Existing System

The existing tesseract-OCR supports English language as
default and also supports languages like Dutch, Spanish, Italian,
French and German. All these languages are trained to
tesseract OCR, but languages like Tamil, Malayalam are not
much trained to OCR. There are no GUI available for tesseract
in tamil and training tesseract is a big task which an
intermediate persons too feel complex for training it. Since
Tesseract OCR Engine is command line Tool, usage of OCR is
much less.

(jaya27) Jaya-com:research JJo$ █
```

Fig. 2. Reading text from an image using OCR.

2.2 NLTK

The NLTK is a library and program for symbolic and statistical natural language processing (NLP) for English written in the Python programming language. It was developed by Steven Bird and Edward Loper in the Department of Computer and Information Science at the University of Pennsylvania. The NLTK supports research and teaching in NLP or closely related areas, including experiential linguistics, cognitive science, artificial intelligence, information retrieval, and machine learning. It has been successfully used as an educational tool, a separate learning tool, and a platform for building research systems. It supports classification, tokenization, morpheme analysis, tagging, parsing, and semantic reasoning. Fig. 3 shows the morphemes of the text “at least nine-tenths of the students passed” in a tree form using the Treebank.

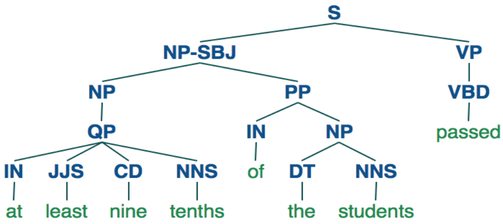


Fig. 3. Display a parse tree.

2.3 GAN

A GAN consists of two models, which are a generative model *G* that approximates the distribution of data and a differential model *D* that distinguishes between real and false. When creating an image using GAN, *D* learns ways of identifying a real image as real and a forged image as a fake image. From the latency vector, *G* learns to create a forged image that is identical to an actual image to deceive the identifier. Fig. 4 shows the GAN structure.

$$\min minV(D,G) = E_{x\sim p_{data(x)}}[logD(x)] + E_{z\sim p_z(z)}[\log(1-D(G(z)))] \tag{1}$$

When learning with an actual image, D must maximize $V(D,G)$. $E_{x\sim p_{data(x)}}$ means sample x in the actual data distribution where $logD(x)$ is maximized when $D(x)$ is 1. $E_{x\sim p_{data(x)}}$ denotes a sample of the potential vector z in the Gaussian distribution. Here, $\log(1-D(G(z)))$ is maximized when $D(G(z))$ is zero. If training with a fake image, the only right part of Eq. (1) is used. On the left part where G must minimize $V(D,G)$, G is not needed; thus, when $D(G(z))$ is 1, $\log(1-D(G(z)))$ is minimized [10,11].

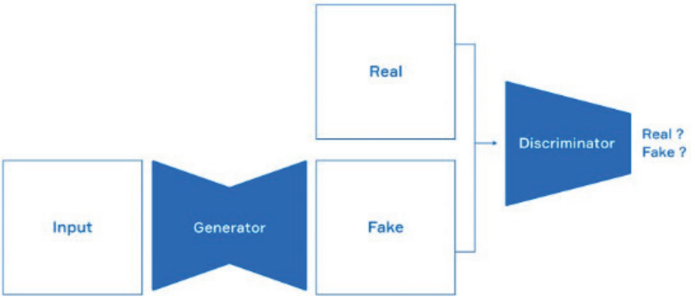


Fig. 4. Principle of learning process of GAN.

2.4 DCGAN

A deep convolutional GAN (DCGAN) was used to overcome the GAN instability. A DCGAN incorporates convolutional structure into the GAN, as shown in Fig. 5. The DCGAN generator structure CNNs are often used in the field of computer vision but are less frequently used in the area of unsupervised learning. The generator learned with DCGAN facilitates vector arithmetic operations, thereby allowing

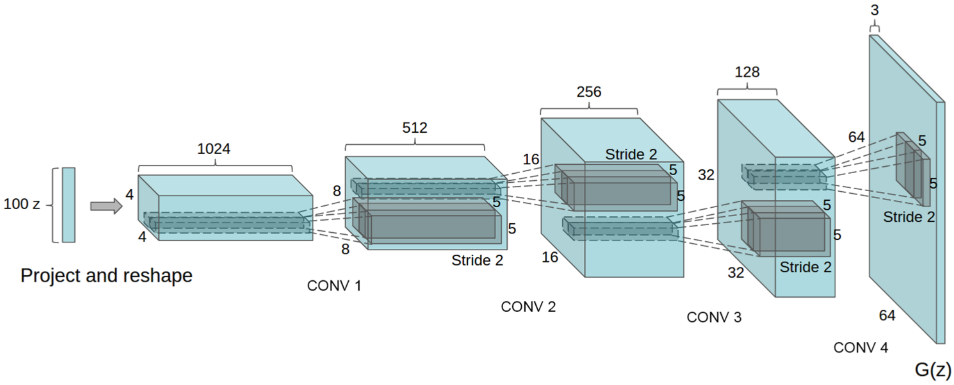


Fig. 5. DCGAN generator structure.

the generation of images at the semantic level [12,13]. For instance, as shown in Fig. 6, in terms of image generation using DCGAN, “glassed woman” is generated as a result of “man with glasses” – “man” + “woman.” The learning model learned glasses, men, and women [14,15].

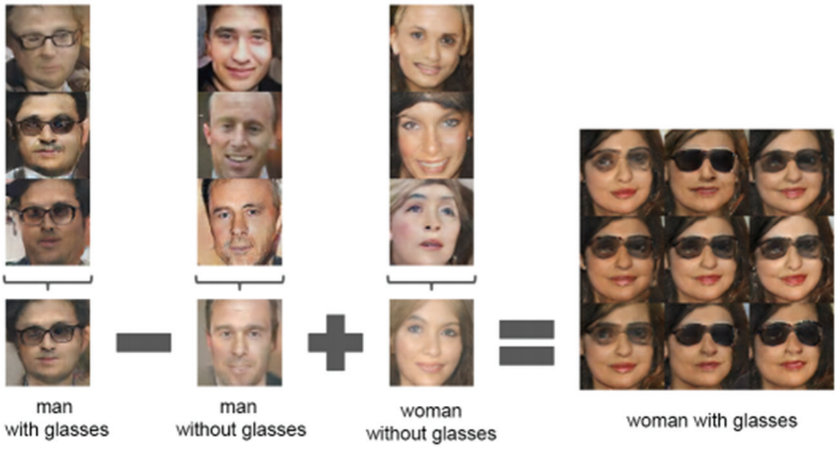


Fig. 6. Image generation using DCGAN.

3. Method

This system included DCGAN training, image-text matching, and the create image process, as shown in Fig. 7. This study system. The DCGAN training step helped in learning the images. The image learning process proceeded to a dataset in ImageNet. The matching title and image using 1,164 images were

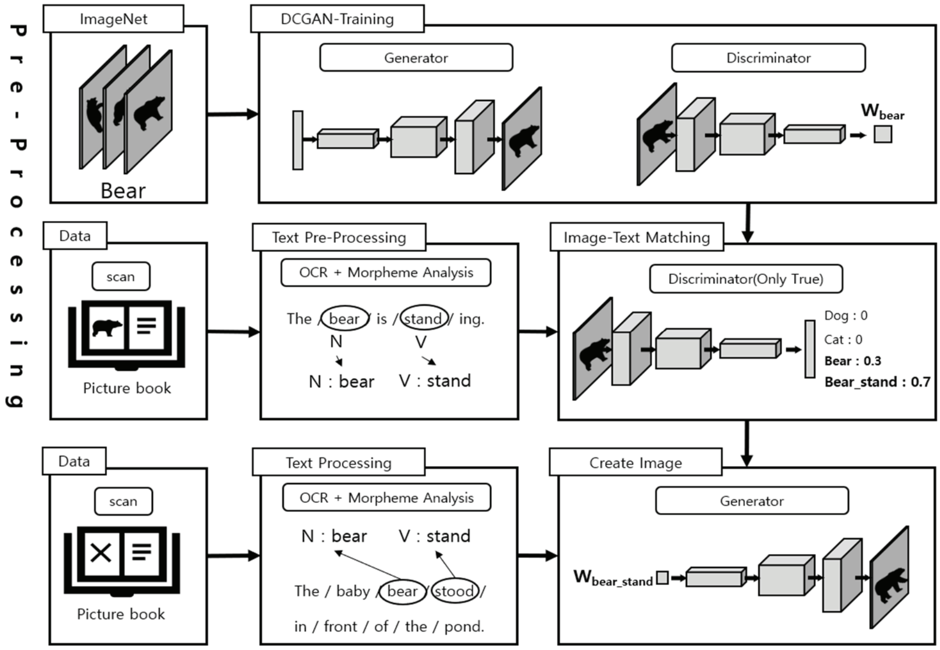


Fig. 7. System design flow chart.

categorized into 11 titles. The image size was 64×64×3, basic learning rate was 0.0002, mini-batch size was 64, epochs were 600, and generator noise was sampled at a 100-dimensional normal distribution. In this process, it learned an alternator between the generator and discriminator.

The image-text matching step matched the text and images from the kids' book. Tesseract, an OCR engine, extracted the images and text from the kids' book. The text additionally underwent morphology analysis to extract nouns and verbs. The discriminator matched the image and text by learning the extracted image as input data and the extracted nouns and verbs as output data.

The image step generated an image. Tesseract extracted text from the kids' book, and it extracted nouns and verbs through morphological analysis. The generator input the extracted text into a latent vector and generated a new image [16,17].

3.1 Dataset

ImageNet is a large visual database designed for visual object recognition software research. It contains more than 20,000 categories, including general categories, which consist of hundreds of images. This study collected bear images with the keyword "bear" for image learning purpose. The wnid "n02131653" in the bear category was used as the key value and downloading was performed using the URL of the images, as shown in Fig. 8. It took 50 minutes to download and receive 1,434 image files. Among these, 1,164 files were used as datasets, while 270 files, which failed to open owing to downloading errors, were excluded.

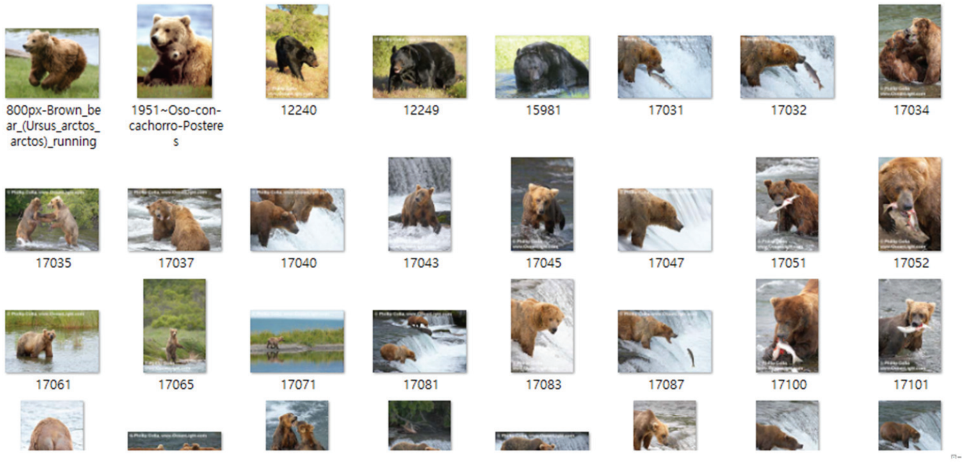


Fig. 8. ImageNet Bear dataset.

3.2 Data Training

A DCGAN subject is trained in terms of the text function encoded by the verb and noun extracted from the kids' book. The generator network G and discriminator network D conditionally perform feed-forward estimation of the text features. The generator is represented as $\boldsymbol{G}: \mathbb{R}^{\boldsymbol{Z}} \times \mathbb{R}^{\boldsymbol{T}} \rightarrow \mathbb{R}^{\boldsymbol{D}}$ and the discriminator is represented as $\boldsymbol{G}: \mathbb{R}^{\boldsymbol{Z}} \times \mathbb{R}^{\boldsymbol{T}} \rightarrow \mathbb{R}^{\boldsymbol{D}}$. Here, $\boldsymbol{T}$ is the size of the part-of-speech, $\boldsymbol{D}$ is the size of the image, and $\boldsymbol{Z}$ is the amount of noise input to $\boldsymbol{G}$. Fig. 9 shows the DCGAN structure discussed in this study.

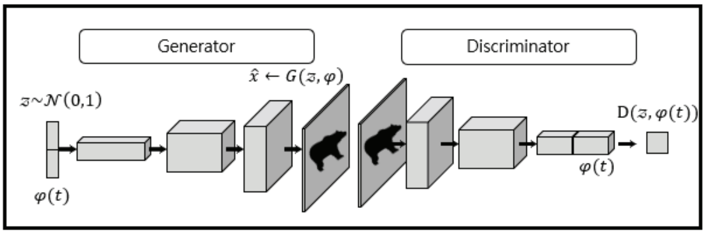


Fig. 9. Basic DCGAN structure of this study.

In G, first, sampling is completed from the noise of $z \in \mathbb{R}^Z \sim \mathcal{N}(0,1)$. By using the text encoder φ, the text query t is encoded. $\varphi(\mathrm{t})$, which contains the part-of-speech, initially compresses the full connection layer using a small dimension (which is 128, to be precise), uses a leaky rectified linear unit (ReLU), and then, finally, connects to the noise vector z. After this inference, it proceeds like a normal deconvolutional network.

Feed-forward is conducted through G and a composite image $\hat{x}$ is generated through $\hat{x} \leftarrow G(z, \varphi(t))$. Image generation corresponds to the feed-forward reasoning of G conditionally applied to the parts-of-speech and noise samples. In D, several layers are performed with the use of leaky ReLU and stride 2 along with batch normalization.

Firstly, improving the accuracy of a class with the classified dataset $\varphi(t)$ is ignored as the part-of-speech is not entered. In the second and third steps, the noun and verb extracted by the text preprocessor become t. The same GAN structure is used for all the datasets. The size of the training image is set to 64×64×3. The text encoder creates a 1024-dimensional embedding projected in 128 dimensions in both the generator and discriminator before increasing the depth with convolutional feature maps.

The generator and discriminator are updated alternatively. With respect to adaptive moment estimation (Adam), an initial learning rate of 0.0002 was used along with Adam's momentum of 0.5. The generator noise was sampled at a 100-dimensional unit normal distribution. A minibatch size of 64 was used, and the training was conducted for 600 epochs. Fig. 10 shows the feature map created by training bear images.

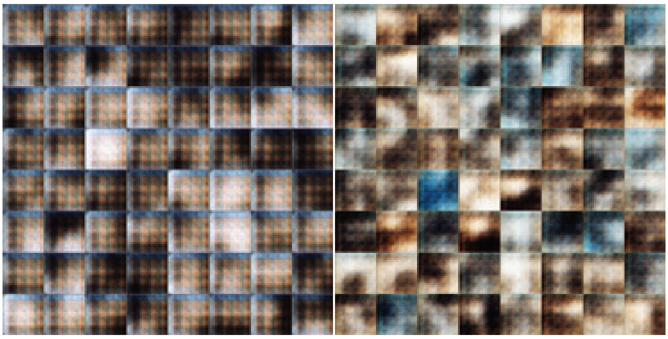


Fig. 10. Training feature map.

3.3 Text Pre-processing

The kids' book used in this study was "Bear Stays Up," which was released on October 5, 2004. As shown in Fig. 11, the study involved the use of the word "But the bear stays up". Further, as shown in Fig. 10, the bear images used had the words "But the bear stays up". The images were photographed using the iPad Pro 10.5-inch second generation. Histogram equalization and sharpening were used to improve text and image recognition after photographing.

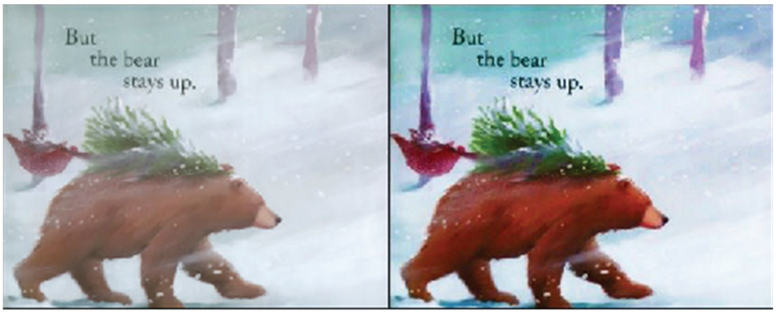
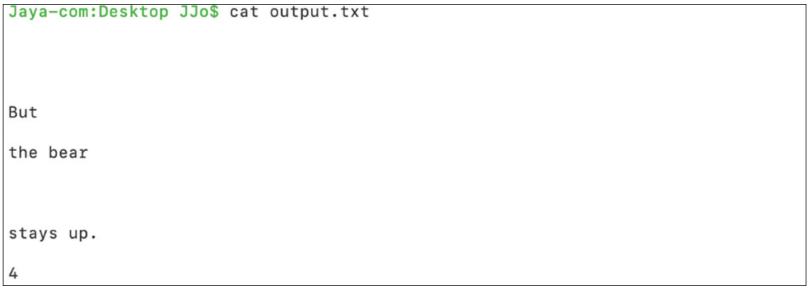


Fig. 11. The kids' book, "But the bear stays up"

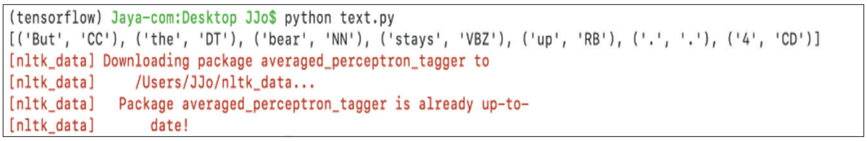


```
Jaya-com:Desktop JJo$ cat output.txt



But

the bear


stays up.

4
```

Fig. 12. Text recognized in the corrected image.

Histogram equalization is the task of uniformly distributing the histogram of an image because when a histogram is significantly concentrated in a specific region, it can result in a low recognition rate. Sharpening is the process of sharpening an image using a filter. It was conducted using a Laplacian filter, and the filter values were {[-1, -1, -1], [-1, 9, -1], [-1, -1, -1]}. The image on the left of Fig. 11 is the original, whereas the one on the right is the corrected version. Fig. 12 shows an image of the letters in the corrected image. The Tesseract used for the extraction shows that the image has a high character recognition accuracy.

The morpheme analysis used the NLTK, which is a Python package for natural language processing and document analysis developed for educational purpose. It has a variety of functions and examples, and it is widely used in practice and research. The two NLTK functions used were tokenization, which removes whitespace and newline characters from a character string with the use of word_tokenize and tagging, which acquired the parts-of-speech of each word with pos_tag. Fig. 13 shows the parts-of-speech of each word from the text ("But the bear stays up") extracted from Fig. 12. Out of these, the NN (noun) and VB (verb) are extracted to be used in DCGAN learning [18].



```
(tensorflow) Jaya-com:Desktop JJo$ python text.py
[('But', 'CC'), ('the', 'DT'), ('bear', 'NN'), ('stays', 'VBZ'), ('up', 'RB'), ('.', '.'), ('4', 'CD')]
[nltk_data] Downloading package averaged_perceptron_tagger to
[nltk_data]     /Users/JJo/nltk_data...
[nltk_data]   Package averaged_perceptron_tagger is already up-to-
[nltk_data]       date!
```

Fig. 13. Tagging result of character string using NLTK.

4. Conclusion

The study proposed text-image matching and image generation using DCGAN to create images not represented in kids' book. By visualizing the text in kids' book, the proposed system can help children use smart devices as educational media.

In total, three factors affect image accuracy. The first factor is camera resolution. The sharper the picture, the higher is the accuracy. The second factor is image correction. In our study, histogram equalization and sharpening were used for easy character recognition in images.

In addition, several other factors such as font size, color difference of the letters, and background required correction of images based on the corresponding situation to obtain good results. The third factor is a large amount of quality data. In data learning, the more the training data, the higher is the accuracy.

In future studies, the data accuracy and object type will be enhanced with the addition of 8,189 flower datasets and 6,033 Caltech-UCSD Birds datasets from the Oxford 102 flowers dataset. With an expansion in scope, utilization would also expand, which can ultimately lead to the creation of images for school textbooks. This would help children and adolescents to understand parts that they cannot comprehend based only on the text provided in the books.

Acknowledgement

This work has supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (No. NRF-2017R1A2B4008886).

References

[1] I. T. Kim and K. J. Yoo, "Effects of augmented reality picture book on the language expression and flow of young children's in picture book reading activities," *The Journal of Korea Open Association for Early Childhood Education*, vol. 23, no. 1, pp. 83-109, 2018.

[2] K. M. Ryu, H. J. Kim, H. J. Kim, E. J. Lee, and J. Y. Heo, "A development of interactive storybook with digital board and smart device," in *Proceedings of the HCI Society of Korea*, Pyeongchang, Korea, 2017, pp. 1179-1182.

[3] Y. Kim and H. Park, "Study on the relation between young children's smart device immersion tendency and their playfulness," *Early Childhood Education Research & Review*, vol. 20, no. 4, pp. 337-353, 2016.

[4] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," *Advances in Neural Information Processing Systems*, vol. 27, pp. 2672-2680, 2014.

[5] R. Tachibana, T. Matsubara, and K. Uehara, "Semi-supervised learning using adversarial networks," in *Proceedings of 2016 IEEE/ACIS 15th International Conference on Computer and Information Science (ICIS)*, Okayama, Japan, 2016, pp. 1-6.

[6] M. S. Ko, H. K. Roh, and K. H. Lee, "GANMOOK: generative adversarial network to stylize images like ink wash painting," in *Proceedings of the Korea Computer Congress*, 2017, pp. 793-795.

[7] L. C. Yang, S. Y. Chou, and Y. H. Yang, "MidiNet: a convolutional generative adversarial network for symbolic-domain music generation," in *Proceedings of the 18th International Society of Music Information Retrieval Conference*, Suzhou, China, 2017, pp. 324-331.

[8] G. C. Lee and J. Yoo, "Development an Android based OCR application for Hangul food menu," *Journal of the Korea Institute of Information and Communication Engineering*, vol. 21, no. 5, pp. 951-959, 2017.

[9] R. Smith, "An overview of the Tesseract OCR engine," in *Proceedings of the 9th International Conference on Document Analysis and Recognition* (ICDAR), 2007, Parana, Brazil, pp, 629-633.

[10] A. C. Rodriguez, T. Kacprzak, A. Lucchi, A. Amara, R. Sgier, J. Fluri, T. Hofmann, and A. Refregier, "Fast cosmic web simulations with generative adversarial networks," *Computational Astrophysics and Cosmology*, vol. 5, article no. 4, 2018.

[11] R. Yamashita, M. Nishio, R. K. G. Do, and K. Togashi, "Convolutional neural networks: an overview and application in radiology," *Insights into Imaging*, vol. 9, no. 4, pp. 611-629, 2018.

[12] A. Radford, L. Metz, and S. Chintala, "Unsupervised representation learning with deep convolutional generative adversarial networks," 2015 [Online]. Available: https://arxiv.org/abs/1511.06434.

[13] Y. Han and H. J. Kim, "Face morphing using generative adversarial networks," *Journal of Digital Contents Society*, vol. 19, no. 3, pp. 435-443, 2018.

[14] S. Reed, Z. Akata, X. Yan, L. Logeswaran, B. Schiele, and H. Lee, "Generative adversarial text to image synthesis," in *Proceedings of the 33nd International Conference on Machine Learning (ICML)*, New York, NY, 2016, pp. 1060-1069.

[15] J. T. Springenberg, A. Dosovitskiy, T. Brox, and M. Riedmiller, "Striving for simplicity: the all convolutional net," 2014 [Online]. Available: https://arxiv.org/abs/1412.6806.

[16] D. Triantafyllidou and A. Tefas, "Face detection based on deep convolutional neural networks exploiting incremental facial part learning," in *Proceeding of 2016 23rd International Conference on Pattern Recognition (ICPR)*, Cancun, Mexico, 2016, pp, 3560-3565.

[17] E. Learned-Miller, G. B. Huang, A. RoyChowdhury, H. Li, and G. Hua, "Labeled faces in the wild: a survey," *Advances in Face Detection and Facial Image Analysis*. Cham, Switzerland: Springer, 2016, pp. 189-248.

[18] Y. Susanti, T. Tokunaga, H. Nishikawa, and H. Obari, "Automatic distractor generation for multiple-choice English vocabulary questions," *Research and Practice in Technology Enhanced Learning*, vol. 13, article no. 15, 2018.

Jaehyeon Cho **https://orcid.org/0000-0002-8255-6875**

He received B.S. degrees in School of Computer Science and Engineering from Hoseo University in 2018, respectively. Since March 2018, he is current with the Department of Computer Science and Engineering from Hoseo University as Master Course.

Nammee Moon **https://orcid.org/0000-0003-2229-4217**

She received B.S., M.S., and Ph.D. degrees in School of Computer Science and Engineering from Ewha Womans University in 1985, 1987 and 1998, respectively. She served as an assistant professor at Ewha Womans University from 1999 to 2003. From 2003 to 2008, she served a professor of Digital Media, Graduate School of Seoul Venture Information. Since 2008, he is currently a professor of Computer Science at Hoseo University. She is current research interests include social learning, HCI and user centric data big data processing and analysis.