

레터논문 (Letter Paper)

방송공학회논문지 제25권 제2호, 2020년 3월 (JBE Vol. 25, No. 2, March 2020)

<https://doi.org/10.5909/JBE.2020.25.2.275>

ISSN 2287-9137 (Online) ISSN 1226-7953 (Print)

## 가상 데이터 생성을 통한 딥러닝 기반 문자인식 시스템 제안

이승주<sup>a)</sup>, 박구만<sup>b)‡</sup>

### Proposal for Deep Learning based Character Recognition System by Virtual Data Generation

Seungju Lee<sup>a)</sup> and Gooman Park<sup>b)‡</sup>

#### 요 약

본 논문에서는 가상 데이터 생성을 통한 딥러닝 기반 문자인식 시스템을 제안한다. 지도학습에서 가장 큰 비중을 차지하는 학습 데이터를 확보하기 위하여 가상 데이터를 생성하였다. 또한 가상 데이터를 생성 후 증강 파라미터를 이용하여, 실제 다양한 데이터에 대응하기 위해서 데이터 일반화를 하였다. 최종적으로 학습 데이터 구성은 증강 파라미터와 폰트 인자에 다양한 값을 대입하여 데이터를 생성하였다. 문자인식 성능을 측정하기 위한 테스트 데이터는 실제 촬영된 이미지 데이터에서 문자영역을 크롭하여 구성하였다. 테스트 데이터는 실제환경에서 발생할 수 있는 이미지 왜곡을 고려하여 데이터 증강하였다. 딥러닝 알고리즘은 실시간 검출에 용이한 YOLO v3를 사용하였으며, 추론결과는 후처리를 통하여 최종 검출결과를 출력한다.

#### Abstract

In this paper, we proposed a deep learning based character recognition system through virtual data generation. In order to secure the learning data that takes the largest weight in supervised learning, virtual data was created. Also, after creating virtual data, data generalization was performed to cope with various data by using augmentation parameter. Finally, the learning data composition generated data by assigning various values to augmentation parameter and font parameter. Test data for measuring the character recognition performance was constructed by cropping the text area from the actual image data. The test data was augmented considering the image distortion that may occur in real environment. Deep learning algorithm uses YOLO v3 which performs detection in real time. Inference result outputs the final detection result through post-processing.

Keyword : YOLO, Virtual Data Generation, Object Detection, Text Recognition

a) 서울과학기술대학교 일반대학원 미디어IT공학과(Dept. of Media IT Engineering, The Graduate Schol, Seoul National University of Science and Technology)

b) 서울과학기술대학교 전자IT미디어공학과(Dept. of Electronic IT Media Engineering, Seoul National University of Science and Technology)

‡ Corresponding Author : 박구만(Gooman Park)

E-mail: [gmpark@seoultech.ac.kr](mailto:gmpark@seoultech.ac.kr)

Tel: +82-2-970-6430

ORCID: <https://orcid.org/0000-0002-7055-5568>

※ This study was supported by the Research Program funded by the SeoulTech(Seoul National University of Science and Technology)

· Manuscript received February 11, 2020; Revised March 18, 2020; Accepted March 18, 2020.

## 1. 서론

딥러닝의 발전으로 다양한 분야에서 딥러닝을 활용한 연구가 활발히 진행중에 있으며, 컴퓨터비전 분야에서 최근 10년 동안 딥러닝을 활용하여 다양한 연구와 성과를 보이고 있다. 특히, 지도 학습(Supervised Learning)을 이용한 딥러닝 분야의 연구가 활발히 진행되면서 데이터의 역할이 큰 비중을 차지하고 있다. 예를 들어, 이미지 데이터 셋은 손글씨 숫자로 구성된 MNIST<sup>[1]</sup>와 1000개 이상의 카테고리 로 구성된 대용량 이미지 데이터 셋인 ImageNet<sup>[2]</sup> 등이 있다. 이러한 검증된 데이터 셋은 딥러닝 알고리즘의 성능을 평가하기 위하여 꼭 필요한 데이터 셋이지만, 좁은 영역의 특정한 작업에서는 학습에 적용하기 힘든 데이터 셋이다. 또한 확보한 데이터의 수가 부족하면, 확보된 데이터 셋을 이용하여 훈련 및 평가를 하는데 힘든 문제점이 있다.

따라서 본 논문에서는 이러한 문제를 해결하기 위하여, 가상 데이터 생성(Virtual Data Generation)을 통한 딥러닝 기반 문자인식 시스템을 제안한다. 가상 데이터 생성을 이용하여 대량의 학습 데이터를 확보하고 이를 기반으로 높은 문자 인식 정확도를 도출하는 것이 목표이다. 가상 데이터 생성의 이점은 객체에 대한 레이블링 작업 및 학습을 위한 별도의 후처리(Post-processing)과정이 필요 없기 때문에 시간과 비용을 절감할 수 있다. 학습 데이터 구성은 모두 가상 데이터만을 사용하며, 모델을 검증하기 위한 테스트 데이터는 실제 환경에서 촬영된 이미지 파일로 구성하였다. 딥러닝 알고리즘은 객체검출 방법 중에서 실시간 객체 검출에 적합한 YOLO v3<sup>[3]</sup>를 사용하였다. 본문에서는 가상 데이터 생성 및 제안한 시스템에 관하여 서술하고 결론에서는 실험 및 실험 분석을 통한 추후 연구방향을 제시한다.

## II. 본론

그림 1은 본 논문에서 제안하는 시스템을 훈련 파트와 추론 파트로 나누어서 표현하였다.

먼저, 훈련 파트(Training Part)에서는 가상 데이터 생성을 통해 획득한 데이터를 다양한 환경에서 일반화될 수 있도록 데이터 증강(Data Augmentation)을 수행한 후 학습을 진행한다.

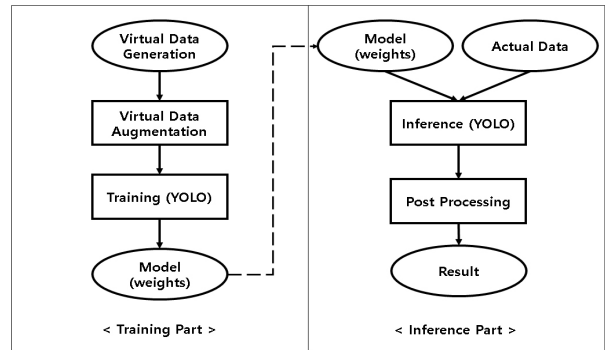


그림 1. 제안하는 시스템 구성도  
Fig. 1. Block diagram of proposed system

이후 추론 파트(Inference Part)에서는 학습 파트의 결과물인 모델(weights)과 실제 촬영된 테스트 데이터(Actual Data)를 추론 모듈에 입력하여 데이터에 대한 결과값을 도출한다. 최종적으로 도출된 데이터를 후처리 후 최종 결과값을 출력한다.

### 1. Virtual Data Generation & Virtual Data Augmentation

가상 데이터 생성은 실제 데이터를 확보하기 어렵거나 레이블링을 포함한 데이터 정리 및 준비에 많은 시간이 소요되는 문제점을 해결하기 위한 방법이다. 가상데이터 생성된 결과는 그림 2(a)와 같으며, 데이터 생성시 두 종류의 글씨체(Bold, Thin Font)를 사용하여 각각 생성하였다. 또한 실제 환경에서 발생할 수 있는 여러 변형을 대비하기 위하여, 생

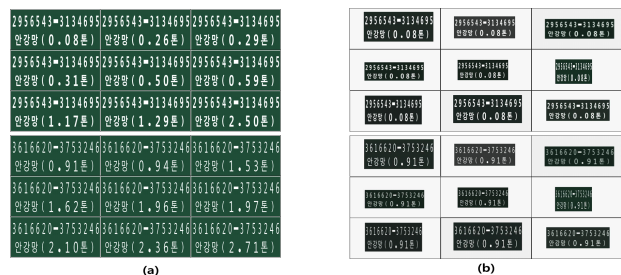

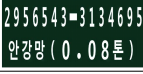

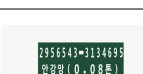
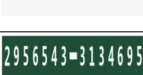
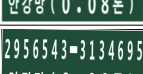


그림 2. 가상 데이터 생성. (a) : 얇은 글씨체와 두꺼운 글씨체 생성 결과, (b) : 데이터 증강 결과  
Fig. 2. Virtual Data Generation. (a): Thin, Bold Font Generation result, (b): Data Augmentation result

표 1. 가상 데이터 증강 매개 변수  
 Table 1. Virtual Data Augmentation Parameter

Parameter Name	Augmentation Result	Parameter Value	Remarks
None		None	Original Image
Contrast		1 ~ 1.5	-
Affine Transformation 1		X : -0.06 ~ 0.06 Y : -0.06 ~ 0.06 Scale X : 0.4 ~ 0.8 Scale Y : 0.4 ~ 0.8	Background black
Affine Transformation 2		X : -0.06 ~ 0.06 Y : -0.06 ~ 0.06 Scale X : 0.4 ~ 0.8 Scale Y : 0.4 ~ 0.8	Background white
GaussianBlur		0 ~ 2.0	-
Rotation		-3 ~ 3	-

성된 가상 데이터 기반으로 증강 파라미터를 적용하여 데이터를 증강하였다. 데이터 증강에 대한 파라미터 및 결과 이미지는 표 1에 표기하였으며, 생성된 결과는 그림 2(b)와 같다.

## 2. Training & Inference parameter

딥러닝 기반 객체 검출 알고리즘은 크게 두 분류로 나뉘지며, 검출 속도는 느리지만 정확도가 높은 2-Stage Detector (R-CNN<sup>[4]</sup>, Fast R-CNN<sup>[5]</sup>, Faster R-CNN<sup>[6]</sup>, etc)와 검출 속도는 빠르지만 정확도가 낮은 1-Stage Detector(YOLO, SSD<sup>[7]</sup>, etc)가 있다. 본 논문에서는 문자인식을 실시간으로 도출하기 위하여 실시간 검출이 가능한 YOLO v3를 사용

하였다. 학습 파라미터는 입력데이터의 크기와 속도의 trade-off를 고려하여 width=416, height=416, channels=3으로 설정하였고, 클래스는 숫자와 한글 그리고 특수문자로 구성하여 classes=83으로 설정하였다.

## 3. Post Processing

YOLO v3의 검출 결과는 검출된 객체 클래스와 좌표(x, y, w, h)로 구성되어 있으며, 각 객체의 클래스 확률을 기준으로 내림차순 정렬되어있다. 따라서 객체의 위치를 기준으로 정렬하기 위해 객체의 중앙점을 계산하고 계산된 중앙점을 기준으로 X축에 대해서 오름차순으로 정렬하여 최종 결과값을 출력하였다.

## III. 실험 결과 및 분석

본 논문에서 제안한 시스템의 학습모델별 성능을 평가하기 위하여 실제 데이터를 기반으로 실험을 진행하였다. 테스트 데이터는 총 200장으로 구성하였으며, 테스트 데이터의 구성은 실제 환경에서 촬영한 데이터를 크롭하여 20장을 획득한 후, 실제 환경에서 발생할 수 있는 이미지 왜곡을 고려하여 데이터를 증강하였다. 학습 데이터는 표 2와 같이 구성하였으며, Model\_0과 Model\_5를 기반으로 데이터를 증강하여 각각의 모델을 생성하였다. 데이터 증강시 사용한 증강 파라미터는 표 2에 각각의 모델에 대해서 표기하였으며, Remarks는 데이터 증강 후 여백을 흰색 또는 검정색으로 처리한 부분을 기록하였다.

실험은 총 200장으로 구성된 테스트 데이터를 10개의 모델로 평가하여 평가결과를 정확도(Accuracy)로 측정하였다. 그림 3은 총 10개의 모델로 평가한 결과를 하나의 그래프로

표 2. 모델별 가상 데이터 증강 정보  
 Table 2. Virtual Data Augmentation information for each model

	Model Name (Bold/Thin Font)	Augmentation Parameter	Number of image data	Remarks
1	<b>Model_0 / Model_5</b>	None	48,216	-
2	<b>Model_1 / Model_6</b>	Contrast, Affine Transformation, GaussianBlur	515,328	Background black
3	<b>Model_2 / Model_7</b>	Contrast, Affine Transformation, GaussianBlur	515,328	Background white
4	<b>Model_3 / Model_8</b>	Contrast, Affine Transformation, GaussianBlur, Rotation	515,328	Background black
5	<b>Model_4 / Model_9</b>	Contrast, Affine Transformation, GaussianBlur, Rotation	515,328	Background white

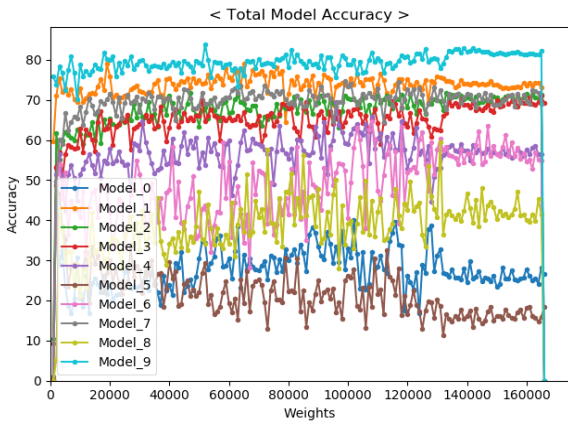


그림 3. 전체 모델의 정확도  
Fig. 3. Accuracy of all models

표현하였다. 그림 4(a)와 그림 4(b)는 그림 3을 두 종류의 글씨체(Bold, Thin Font)로 나누어 각각 그래프로 표현하였다.

전반적으로 얇은 글씨체로 학습된 모델들이 좋은 결과를 보여주고 있다. 특히 그림 4(b)에 Model\_9는 본 실험에서 가장 높은 결과를 보여주고 있다. Model\_9가 가장 높은 결과를 보여주고 있는 이유는 학습 데이터에 회전을 적용한 데이터가 포함되어 있으며, 테스트 데이터 생성과정에서 데이터 증강 후 여백을 흰색으로 표현했기 때문에 좋은 결과를 보여주는 것으로 분석된다.

본 실험에서는 학습 데이터를 생성할 때, 얇은 글씨체와 회전 그리고 데이터 여백의 색상이 문자 인식에 영향을 미치는 요소임을 확인하였다.

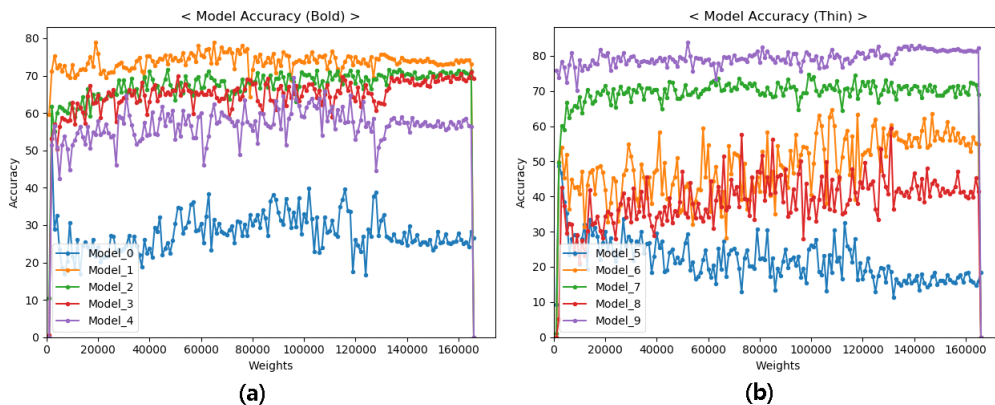


그림 4. 글씨체 차이에 따른 모델의 정확도. (a) : 굵은 글씨체 모델 정확도, (b) : 얇은 글씨체 모델 정확도  
Fig. 4. Accuracy of model according to font difference. (a) : Bold Font model Accuracy, (b) : Thin Font model Accuracy

#### IV. 결론

본 논문에서 가상 데이터 생성을 통한 딥러닝 기반 문자 인식 시스템을 제안하기 위하여 가상 데이터 생성 방법과 학습 및 추론 파라미터, 후처리에 대해 소개하였다. 그리고 제안한 시스템 성능을 측정하기 위하여 얇은 글씨체와 굵은 글씨체로 학습한 모델에 대해서 실험 및 분석하였다. 본 논문에서 실험한 결과를 바탕으로, 추후에는 문자 인식 정확도 개선방법과 전체 이미지에서 문자 영역을 찾은 후 문자를 인식하는 시스템에 대해서 연구할 예정이다.

#### 참고 문헌 (References)

- [1] Deng, Li, "The mnist database of handwritten digit images for machine learning research [best of the web]," *IEEE Signal Processing Magazine*, Vol.29, No.6, pp.141-142, November 2012, doi:10.1109/MSP.2012.2211477.
- [2] Russakovsky, Olga, et al, "Imagenet large scale visual recognition challenge," *International journal of computer vision*, Vol.115, No.3, pp.211-252, December 2015, doi:10.1007/s11263-015-0816-y.
- [3] Joseph Redmon and Ali Farhadi, "YOLOv3: An Incremental Improvement," arXiv preprint arXiv:1804.02767, 2018, https://arxiv.org/abs/1804.02767.
- [4] Girshick, Ross, et al., "Rich feature hierarchies for accurate object detection and semantic segmentation," *Proceedings of the IEEE conference on computer vision and pattern recognition*, Ohio, United States of America, pp. 580-587, 2014.
- [5] GIRSHICK, Ross, "Fast r-cnn," *Proceedings of the IEEE international conference on computer vision*, Santiago, Chile, pp. 1440-1448, 2015.
- [6] REN, Shaoqing, et al, "Faster r-cnn: Towards real-time object detection with region proposal networks," *Advances in neural information processing systems*, Montreal, Canada, pp. 91-99, 2015.
- [7] LIU, Wei, et al., "Ssd: Single shot multibox detector," *European conference on computer vision*, Springer, Amsterdam, Netherlands, pp. 21-37, 2016, doi:10.1007/978-3-319-46448-0\_2.