

# 인셉션 모듈 기반의 보다 깊은 컨볼루션 신경망을 통한 한글 필기체 인식

강우영<sup>○</sup> 김병희 장병탁  
 서울대학교 컴퓨터공학부  
 {wykang, bhkim, btzhang}@bi.snu.ac.kr

## Hangeul Handwriting Recognition using Deeper Convolutional Neural Networks Based on Inception Modules

Woo-Young Kang<sup>○</sup>, Byoung-Hee Kim, Byoung-Tak Zhang  
 School of Computer Science and Engineering, Seoul National University

### 요 약

이미지 인식 분야에서 뛰어난 성능을 보인 인셉션 모듈(inception module) 기반의 컨볼루션 신경망(CNN)을 한글 필기체 인식문제에 적용하였다. 깊은 구조를 갖는 신경망은 기하급수적으로 변수가 증가하는 문제가 발생하는데, 이는 인셉션 모듈을 통해 적절하게 통제할 수 있다. 특히 깊은 신경망의 빠른 학습을 위해 최근 발표된 Batch Normalization (BN) 기법을 함께 사용하였다. 이러한 특성을 이용하여 모델을 구성하면, 알파벳이나 숫자에 비해 복잡한 패턴을 보이며 클래스의 수가 매우 많은 한글 필기체를 효과적으로 학습 및 인식할 수 있다. 520클래스, 260,000글자로 구성된 한글 필기체 이미지 DB에 대한 인식실험을 수행한 결과, 인셉션 모듈 중심으로 구성된 컨볼루션 신경망은 비슷한 깊이의 전통적인 컨볼루션 신경망에 필요한 변수의 34.4%정도만으로도 인식률이 1.5%향상됨을 확인하였다.

### 1. 서 론

필기체인식에 관한 연구는 90년대부터 자동화된 우편 번호인식, 문서분류 등 다양한 목적을 위해 본격적으로 연구되기 시작했다. 이러한 필기체 인식 연구는 주로 과거에는 글자를 구성하는 획의 특징을 추출하여 확률 그래프기반의 모델[1]이나 SVM[2] 등과 같은 방법을 사용하여 수행되어왔으며, 숫자 이미지 인식을 위해 얇은 깊이의 컨볼루션 신경망이 사용되기도 하였다[3]. 반면, 최근에는 딥러닝 기술의 등장과 함께 심층 컨볼루션 신경망이나 순환 신경망 등이 필기체 인식에 사용되면서 기존의 알고리즘에 비해 큰 성능 향상을 이루어 냈다 [4][5][6]. 본 논문에서는 딥러닝을 offline 방식의 한글 필기체 낱글자 인식에 적용하는 연구를 수행하였다. 한글 필기체의 경우 복잡한 패턴을 가지며 클래스의 수도 매우 많으므로 숫자나 알파벳보다 인식하기 어렵다고 알려져 있다. 최근 한글 필기체 인식 문제에 깊은 구조를 가지는 컨볼루션 신경망을 통해 인식 정확도를 높인 사례가 보고되었다[7][8]. 하지만, 컨볼루션 신경망은 데이터가 충분히 많을 경우 깊게 쌓을수록 일반적으로 인식 성능이 좋아지는 장점이 있으나, 그만큼 학습시켜야 할 변수들의 수도 기하급수적으로 늘어난다는 단점도 존재한다. 이러한 기하급수적인 변수공간의 확장은 최근 Large Scale Visual Recognition Challenge에서 변수 공간의 크기를 획기적으로 줄이면서도 뛰어난 인식 성능을 보인 인셉션 모듈의 도입을 통해 어느 정도 완화됨을 볼 수 있었다[9]. 또한 Batch Normalization 기법을 활용하면 학습 오류율의 빠른 수렴을 통해 학습을 좀 더

가속화 시킬 수 있다[10]. 본 논문에서는 이러한 인셉션 모듈을 사용하여 가벼우면서도 깊은 컨볼루션 신경망을 구성하고 한글 필기체 낱글자 이미지 분류에 적용한 결과를 보고한다. 실험 데이터로 빈번하게 사용되는 520가지 한글 낱글자에 대한 필기체 이미지를 사용하였으며, 각 글자별 500장씩 총 260,000장의 이미지로 구성되어 있다. 본 실험을 통해 기존의 CNN방식보다 훨씬 적은 변수공간의 크기에도 더 나은 인식 성능을 달성할 수 있음을 확인할 수 있었다.

### 2. 인셉션 모듈을 이용한 깊은 신경망

학습시킬 데이터가 충분히 많을 때 더욱 고수준의 특징을 통한 높은 인식 성공률을 얻기 위해서는 일반적으로 컨볼루션 층을 깊이 쌓는다. 하지만 깊은 층을 쌓음으로써 몇 가지 문제가 발생하는데, 대표적으로 과적합 문제와 gradient vanishing, 그리고 변수공간의 기하급수적인 증가를 들 수 있다. 이 중 앞의 두 문제는 각각

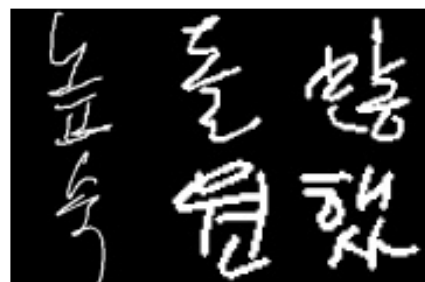


그림 1. 한글 필기체 데이터의 예시

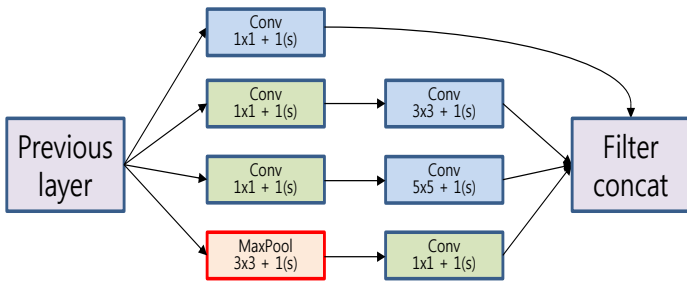


그림 2. 인셉션 모듈의 구조

dropout기법의 사용 [11], 활성화 함수로 ReLU( $ReLU(x) = \max(x, 0)$ )의 사용 등을 통해 완화될 수 있다. 세 번째 문제의 경우 인셉션 모듈 [9]을 통해 효과적으로 통제될 수 있다. 그림 2에서 보듯 한 층의 인셉션 모듈의 구조는 기존의 컨볼루션 층보다 좀 더 복잡하지만 3x3, 5x5 마스크의 깊이를 1x1 마스크 필터의 개수로 통제함으로써 결과적으로 여러 층의 표현을 한 층의 더 적은 변수 집합으로 표현할 수 있다.

### 3. 다층 신경망 모델에서의 Batch Normalization

Batch normalization 기법은 mini-batch 단위로 들어오는 입력 데이터들과 그 데이터들이 각 층을 통과할 때 가지는 분포를 적절하게 조정함으로써 오류율에 대한 수렴을 가속화 시키는 방법이다 [10]. 위에서 언급한 대로 각 층의 입력 데이터 마다 분포가 일정하지 않은 현상을 internal covariate shift라고 하며 각 특징 차원이 독립적임을 가정하고 이를 줄임으로써 학습의 성능 수렴을 가속화 시킬 수 있다. Normalization 과정은 식 (1)(2)와 같다:

$$\hat{x}_i \leftarrow \frac{x_i - \mu_B}{\sqrt{\sigma_B^2 + \epsilon}} \quad (1)$$

$$y_i \leftarrow \gamma \hat{x}_i + \beta \equiv BN_{\gamma, \beta}(x_i) \quad (2)$$

수식 (1)에서  $x_i$ 는 입력 값을 의미하고  $\mu_B$ 와  $\sigma_B^2$ 는 각각 mini-batch의 평균과 분산을 나타내며  $\epsilon$ 는 수식의 분모가 0이 되는 것을 방지하기 위해 넣어주는 작은 수이다. 수식 (2)에서  $\gamma$ 와  $\beta$ 는 학습시켜야 할 변수이다. 테스트 시에는 데이터의 mini-batch에 종속적이지 않게 하기 위해 batch별 평균과 분산을 가중치와 함께 고려한 moving average 기법을 사용해야 한다. 미분 가능한 loss 함수  $\ell$

표 1. 기존의 컨볼루션 신경망을 사용한 한글인식기의 구조

Layers	# of feature maps / stride	# of parameters
Conv1	5x5x32 / 1	832
Pool1	2x2x32 / 2	-
Conv2	5x5x64 / 1	51,264
Pool2	2x2x64 / 2	-
Conv3	4x4x128 / 1	131,200
Pool3	2x2x128 / 2	-
Conv4	4x4x256 / 1	524,544
Pool4	2x2x256 / 1	-
FC1	256x384	98,688
FC2	384x520	200,200
Total	-	1,006,728

이 정해졌을 때 batch normalization이 들어간 층의 오류 역전파는 다음과 같다.

$$\frac{\partial \ell}{\partial \hat{x}_i} = \frac{\partial \ell}{\partial y_i} \cdot \gamma \quad (3)$$

$$\frac{\partial \ell}{\partial \sigma_B^2} = \sum_{i=1}^m \frac{\partial \ell}{\partial \hat{x}_i} \cdot (x_i - \mu_B) \cdot \frac{-1}{2} (\sigma_B^2 + \epsilon)^{-3/2} \quad (4)$$

$$\frac{\partial \ell}{\partial \mu_B} = \left( \sum_{i=1}^m \frac{\partial \ell}{\partial \hat{x}_i} \cdot \frac{-1}{\sqrt{\sigma_B^2 + \epsilon}} \right) + \frac{\partial \ell}{\partial \sigma_B^2} \cdot \frac{\sum_{i=1}^m -2(x_i - \mu_B)}{m} \quad (5)$$

$$\frac{\partial \ell}{\partial x_i} = \frac{\partial \ell}{\partial \hat{x}_i} \cdot \frac{-1}{\sqrt{\sigma_B^2 + \epsilon}} + \frac{\partial \ell}{\partial \sigma_B^2} \cdot \frac{\sum_{i=1}^m -2(x_i - \mu_B)}{m} + \frac{\partial \ell}{\partial \mu_B} + \frac{1}{m} \quad (6)$$

$$\frac{\partial \ell}{\partial \gamma} = \sum_{i=1}^m \frac{\partial \ell}{\partial y_i} \cdot \hat{x}_i \quad (7)$$

$$\frac{\partial \ell}{\partial \beta} = \sum_{i=1}^m \frac{\partial \ell}{\partial y_i} \quad (8)$$

### 4. 실험

같은 깊이의 기존 CNN과 인셉션 모듈을 적용한 CNN 모델 간의 성능 비교 실험을 구성하여 보고한다. 실험에 사용한 데이터는 총 260,000자의 한글 필기체 낱글자 이미지(60\*60 픽셀)로 글자당 500개, 총 520 클래스로

표 2. 인셉션 모듈 기반의 컨볼루션 신경망을 사용한 한글인식기의 구조

Layers	Output	1x1	3x3 reduce	3x3	5x5 reduce	5x5	Pool proj	# of parameters
Incep1	30x30x56	1x16	1x24+24	9x24x16	1x8+8	25x8x8	1x16	5,152
Incep2	15x15x120	1x56x32	1x56x32+32	9x32x32	1x56x8+8	25x8x24	1x56x32	19,880
Incep3	8x8x224	1x120x32	1x120x32+32	9x32x64	1x120x16+16	25x16x64	1x120x64	61,360
Incep4	1x1x256	1x224x64	1x224x48+48	9x48x64	1x224x24+24	25x24x64	1x224x64	110,920
FC1	256*192	-	-	-	-	-	-	49,152
FC2	192*520	-	-	-	-	-	-	99,840
Total	-	-	-	-	-	-	-	346,304

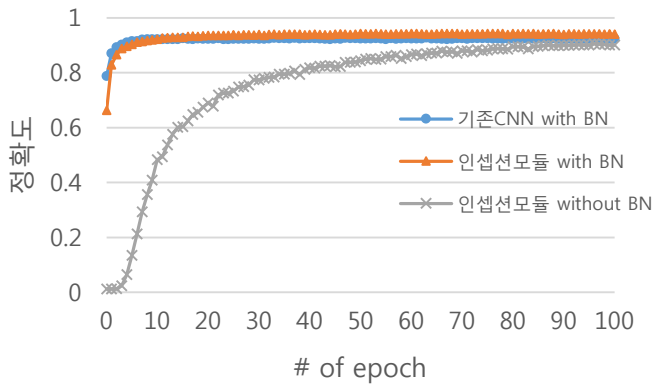


그림 3. 기존의 CNN과 인셉션 모듈에 기반한 CNN의 인식 성능 비교

구성되어 있으며([7]에서 적용한 SERI95a DB와 동일한 클래스 구성) 학습 셋과 테스트 셋의 비율은 4:1로 구성하였다. 학습 시 mini-batch 방식으로 학습을 수행하였고 mini-batch의 크기는 200으로 하였다. 실험에서 인셉션 모듈을 사용한 컨볼루션 신경망은 각 인셉션 모듈 이후 2x2, 2stride max pooling 연산을 수행하였다. 또한 batch normalization은 각 인셉션 모듈의 filter concatenation 이전 4종류의 특징 맵에 대해 각각 수행하였다. 각 모델의 FC2층은 softmax층으로 구성하였으며, 모델의 loss 함수로는 multiclass log loss 함수를 사용하였다. 실험결과 기존의 컨볼루션 신경망은 100번째 epoch 시점에서 92.6%의 인식성능을 보였으며, 인셉션 모듈 기반의 컨볼루션 신경망은 같은 층의 기존 컨볼루션 신경망을 구성하는 데 필요한 변수의 34.4%정도의 변수만을 사용했음에도 불구하고 1.5%높은 94.1%의 인식 성능을 보였다. 또한 batch normalization 기법을 사용한 경우 그렇지 않았을 때보다 인식성능의 향상속도가 매우 가속화됨을 확인할 수 있었다.

5. 결론

본 논문에서는 보다 깊은 컨볼루션 신경망을 통해 한글 필기체 낱글자에 대한 인식을 수행하였다. 컨볼루션 신경망을 깊이 쌓음에 따라 발생하는 변수공간의 증가는 인셉션 모듈을 통해 효과적으로 통제하였으며 batch normalization 기법을 통해 학습 오류율의 수렴을 가속화

표 3. 한글 필기체 인식 결과

Setting	# of epoch	Accuracy	Maximum Accuracy
CNN with BN	100	92.6%	92.7%
Inception module with BN	100	94.1%	94.3%
Inception module without BN	100	90.7%	90.7%

시켰다. 그 결과 기존의 컨볼루션 신경망에 비해 인셉션 모듈을 활용한 컨볼루션 신경망이 훨씬 적은 수의 변수를 사용하면서도 더 나은 인식성능을 보임을 확인할 수 있었다.

감사의 글

이 논문은 2016년도 삼성전자의 지원을 받아 수행된 연구이며, 정부(미래창조과학부)의 재원으로 정보통신기술진흥센터(R0126-16-1072-SW스타랩)와 한국산업기술평가관리원(10044009-HRI.MESSI, 10060086-RISF)의 지원을 일부 받았음.

참고문헌

- [1] Cho, S. J., et al. "Bayesian network modeling of hangul characters for online handwriting recognition." In *Proc. of the IEEE International Conference on Document Analysis and Recognition*, 2003.
- [2] Bahlmann, C., et al. "Online handwriting recognition with support vector machines-a kernel approach." In *Proc. of the IEEE International Workshop on Frontiers in Handwriting Recognition*, 2002.
- [3] LeCun, Y., et al. "Gradient-based learning applied to document recognition." *Proceedings of the IEEE*, 86(11):2278-2324, 1998.
- [4] Sermanet, P., et al. "Convolutional neural networks applied to house numbers digit classification." In *Proc. of the IEEE International Conference on Pattern Recognition*, 2012.
- [5] Liwicki, M., et al. "A novel approach to on-line handwriting recognition based on bidirectional long short-term memory networks." In *Proc. of the IEEE International Conference on Document Analysis and Recognition*, 2007.
- [6] Yin, F., et al. "Icdar 2013 chinese handwriting recognition competition". In *Proc. of IEEE International Conference on Document Analysis and Recognition*, pp. 1464-1470, 2013
- [7] Kim, I. J., et al. "Handwritten Hangul recognition using deep convolutional neural networks". *International Journal on Document Analysis and Recognition*, 18(1):1-13. 2015.
- [8] Kim, I. J., et al. "Improving discrimination ability of convolutional neural networks by hybrid learning". *International Journal on Document Analysis and Recognition*, 19(1):1-9. 2016.
- [9] Szegedy, C., et al. "Going deeper with convolutions". In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1-9, 2015.
- [10] Ioffe, S., et al. "Batch normalization: Accelerating deep network training by reducing internal covariate shift". *arXiv preprint arXiv:1502.03167*. 2015.
- [11] Hinton, G. E., et al. "Improving neural networks by preventing co-adaptation of feature detectors." *arXiv preprint arXiv:1207.0580*. 2012.