

# CNN 기반 이미지 분류 모델 훈련 시 병렬 처리 기술이 학습 속도에 미치는 영향

한예찬, 손우진, 김재윤\*  
순천향대학교

ychan.sch@gmail.com, swj000317@naver.com, \*kimym38@sch.ac.kr

## The effect of parallel processing technology on a CNN-based image classifier

Han Yechan, Shon Woojin, Kim Jaeyun\*  
Soonchunhyang Univ.

### 요 약

컴퓨팅 기술의 발전과 CPU, GPU 등 컴퓨터 연산 장치 성능의 향상은 현대의 다양한 문제들에 대한 통찰력을 제공해 주었다. 특히, 인공지능 분야에서 GPU는 딥러닝 모델의 학습에 사용되어 이미지 분류, 음성 인식 등의 복잡한 문제를 해결하는데 사용되었으며 병렬 처리 기술을 통해 보다 빠른 학습을 수행하였다. 하지만 특정한 상황에서는 GPU의 병렬 처리 기술이 학습 속도를 저하시킬 수 있다. 본 논문에서는 GPU의 병렬 처리 기술이 모델의 학습 속도에 미치는 영향을 살펴보기 위해 CNN 기반 이미지 분류 모델을 예제 데이터 셋으로 훈련시켜 모델의 학습 효율성을 확인했다.

### 1. 서 론

컴퓨팅 기술이 나날이 발전함에 따라 컴퓨터의 연산과 데이터 처리를 담당하는 CPU (Central Processing Unit)의 성능 또한 비약적으로 상승하였으며 컴퓨팅 자원 및 성능 등의 한계로 이전에 다룰 수 없었던 데이터를 처리할 수 있게 되었다[1]. 특히, 인공지능 분야에서 CPU는 여러가지 복잡한 통계식으로 작성된 머신러닝 모델을 순차적으로 처리하기 매우 적합했고 높은 CPU 성능 (큰 클럭, 많은 코어 개수, 큰 캐시메모리 등)은 머신러닝 모델의 빠른 학습을 가능케 했다. 하지만, 이미지 분류, 음성 인식 등의 복잡한 문제들에 대해 신경망 (Neural Network) 기반의 딥러닝 모델들의 성능이 머신러닝 모델을 능가하는 모습을 보여주자 연구자들은 딥러닝 모델의 빠른 학습을 위해 병렬 처리 작업을 수행할 수 있는 GPU (Graphics Processing Unit)를 모델 학습에 사용하게 되었다. GPU는 클럭의 크기가 CPU에 비해 작지만 코어의 수가 매우 많아서 단순하고 반복적인 대량의 연산을 처리하기 적합하다는 특징이 있다[2]. 따라서, 신경망 기반 딥러닝 모델에서 반복적이고 비슷한 대량의 연산 (단순 사칙연산, 벡터나 행렬의 곱셈 연산 등)을 수행할 때 병렬 처리가 가능한 GPU를 사용함으로써 CPU에 비해 더 빠른 연산을 수행할 수 있게 되었다[3]. 최근에는 컴퓨터에 GPU를 여러 개 연결하여 사용하는 SLI (Scalable Link Interface) 기술을 통해 단일 GPU에 비해 더욱 더 빠른 학습을 추구하기도 한다. 하지만 GPU를 이용한 병렬 처리 기술이 항상 좋은 것은 아니다. 병렬 처리 기술은 복잡한 문제에 한정하여 사용해야 하며 단순한 문제에 사용하는 것은 비효율적이다. 본 논문에서는 CNN 기반 딥러닝 모델 학습 시 CPU와 GPU를 사용하며 3개의 예제 이미지 데이터 셋과 3개의 이미지 분류 모델로 학습 효율성을 확인한다.

### 2. 본 론

#### 2.1. 실험 데이터

본 논문에서는 MNIST (숫자 0~9 이미지 분류), CIFAR10 (비행기, 자동차, 새, 고양이, 사슴, 개, 개구리, 말, 배, 트럭 이미지 분류), Cats vs Dogs (개와 고양이 이미지 분류) 데이터 셋을 사용하였으며 각 데이터셋에 대한 설명은 Table 1과 같다.

Table 1 실험 데이터 셋 명세

Dataset	Instances	Classes	Image shape
MNIST	70000	10	(28, 28, 1)
CIFAR10	60000	10	(32, 32, 3)
Cats vs Dogs	25000	2	(128, 128, 3)

#### 2.2. 실험 모델 및 실험 방법

본 논문에서는 이미지 분류를 위한 모델로 CNN, ResNet50, VGG16 모델을 사용하였다. 각 모델의 마지막 출력층은 학습에 사용되는 데이터셋의 라벨 개수 (Classes)로 설정되었으며 ResNet50과 VGG16은 weights를 None으로 설정하여 사전학습 가중치를 가져오지 않고 전체 파라미터를 학습시켰다. Epoch는 각각의 학습마다 10으로 설정하였으며 1 epoch당 평균 학습 속도를 측정하였다. 또한, 모델은 CPU (Intel Core i9-10900X @ 3.70GHz), GPU (GeForce RTX 2080 Ti), Multi-GPU (GeForce RTX 2080 Ti \* 3) 3가지 방법으로 학습되었다.

### 2.3. 실험 결과

Fig 1~3 의 숫자 값은 평균 학습 속도 (초)를 나타내며 데이터 셋 별로 결과를 구분하였다. Fig 1 에서 CPU 를 이용한 CNN 이미지 분류 모델 학습 속도가 GPU, Multi-GPU 를 이용한 것보다 빠른 것을 볼 수 있다. 이는 MNIST 이미지의 크기가 작고 CNN 모델도 단순한 구조이기 때문에 CPU 만으로도 충분히 학습할 수 있기 때문이다. 하지만 ResNet50 모델은 1 epoch 당 970.9 초, VGG16 모델은 1201.6 초로 CPU 를 이용한 모델 학습 속도가 굉장히 느린 것을 확인할 수 있다. 이는 ResNet50, VGG16 모델이 CNN 모델에 비해 매우 복잡한 구조를 가지고 있기 때문이다. Fig 2, Fig 3 에서도 앞선 결과와 마찬가지로 CPU 를 통해 학습하는 경우 ResNet50, VGG16 모델의 학습 속도가 CNN 모델에 비해 느린 것을 확인할 수 있다.

ResNet50 모델과 VGG16 모델을 GPU 와 Multi-GPU 로 학습하는 경우를 보면 MNIST (Fig 1)와 CIFAR10 (Fig 2)에서 ResNet50 모델의 학습 속도는 VGG16 모델보다 느린 것을 볼 수 있다. 이는 ResNet50 모델이 CPU 로 학습할 때는 VGG16 모델 보다 빠르지만 GPU 로 학습할 때는 느려서 나타나는 결과이다[4]. Cats vs Dogs (Fig 3)에서는 반대의 결과가 나타나는데 MNIST, CIFAR10 의 이미지 크기가 Cats vs Dogs 보다 큰 것으로 보아 이는 학습 데이터 셋의 이미지 크기와 연관이 있을 것으로 예상된다.

연산 장치의 관점에서 결과를 확인하면 학습 속도는 이미지의 크기가 가장 큰 Cats vs Dogs 데이터 셋을 기준으로 GPU 가 CPU 에 비해 모델에 따라 약 20 ~ 250 배 정도 빠른 것을 볼 수 있으며 Multi-GPU (동일한 GPU 3 개)가 GPU 에 비해 약 2 배 정도 빠른 것을 볼 수 있다. Multi-GPU 를 사용하였을 때 2 배 정도의 성능 향상만 있던 이유는 GPU 별 메모리 할당 과정과 연산 결과 병합 과정에서 오버헤드로 인한 문제가 발생하기 때문이다[5].

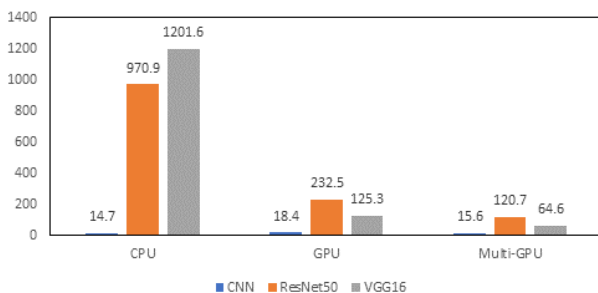


Fig 1 MNIST 이미지 분류 모델의 평균 학습 속도

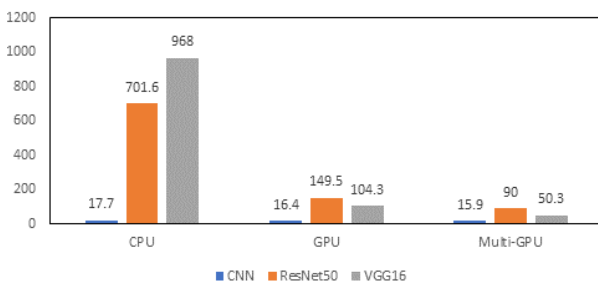


Fig 2 CIFAR10 이미지 분류 모델의 평균 학습 속도

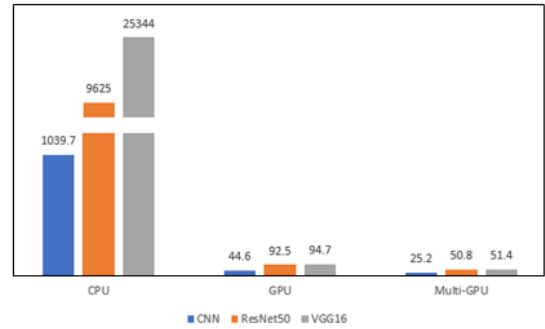


Fig 3 Cats vs Dogs 이미지 분류 모델의 평균 학습 속도

### 3. 결론

본 논문에서는 CNN 기반 이미지 분류 모델 훈련 시 병렬 처리 기술이 모델의 학습 속도에 미치는 영향에 대해 확인했다. CPU, GPU, Multi-GPU 를 활용하여 CNN, ResNet50, VGG16 모델을 훈련시켰으며 MNIST, CIFAR10, Cats vs Dogs 예제 데이터를 이용하여 학습 효율성을 확인하였다. ResNet50, VGG16 과 같이 복잡한 구조를 가진 모델을 GPU 를 사용하여 학습했을 때 학습 속도가 CPU 보다 매우 빠른 것을 볼 수 있었다. 하지만, CNN 과 같이 단순한 구조를 갖는 모델은 특정 상황에서 GPU 보다 CPU 를 사용하는 것이 더 효율적이라는 것을 확인했으며 이는 항상 GPU 를 사용하는 것이 효율적인 것은 아니라는 것을 의미한다. 본 논문은 모델의 종류와 예제 데이터 셋의 개수, 모델 학습 시 설정한 총 Epoch 의 크기가 작아 실험의 결과가 편향되었을 수 있다는 한계점이 있다. 따라서, 유용성을 검증하기 위해 사용하는 모델의 종류와 예제 데이터 수, Epoch 의 크기를 늘려 실험을 더 진행할 수 있다.

### ACKNOWLEDGMENT

본 연구는 2021 년 과학기술정보통신부 및 정보통신기획평가원의 SW 중심대학사업의 연구결과로 수행되었음 (2021-0-01399).

### 참 고 문 헌

- [1] 최성수, 양정모, 이준엽. (2021). 병렬컴퓨팅 발전에 따른 계산과학 분야들의 발전과 확장, 그리고 연구 패러다임 변화(보고서 번호: 2586-1131). 대전: 한국연구재단.
- [2] Owens, J. D., Houston, M., Luebke, D., Green, S., Stone, J. E., & Phillips, J. C. (2008). GPU computing. *Proceedings of the IEEE*, 96(5), 879-899.
- [3] 안신영, 박유미, 임은지, & 최완. (2016). 딥러닝 분산처리 기술동향. [ETRI] 전자통신동향분석, 31(3), 0-0.
- [4] Keras Applications. (n.d.). Retrieved from <https://keras.io/api/applications>.
- [5] 국중진. (2019). 다중 GPU 기반 홀로그램 생성을 위한 병렬처리 성능 최적화 기법. *스마트미디어저널*, 8, 1-7.