특집원고

자율주행을 위한 심층신경망 시스템 최적화 기술 동향

국민대학교 ▌안솔·김종찬*

1. 서 론

최근 몇 년 동안 딥 뉴럴 네트워크(DNN)는 인공지능 분야에서 눈부신 발전을 이루었다. 이러한 기술의 발전은 데이터 분석, 이미지 및 음성 인식, 자연어 처리 등 다양한 분야에서 혁신을 가져와 우리 일상생활에 깊이 자리 잡았다. DNN의 이러한 진화는 고도화된 알고리즘과 빅 데이터, 강력한 컴퓨팅 파워의 결합으로 가능해졌다.

DNN의 활용은 고성능 컴퓨팅 환경에만 국한되지 않는다. 최근에는 모바일 기기, 웨어러블 디바이스, 자동차 등 임베디드 시스템에서도 활발히 사용되고 있다[1]. 특히 자동차 산업에서의 DNN 활용은 주목할 만하다. 자율주행 차량은 DNN을 활용해 주변 환경을 인식하고, 복잡한 도로 상황에서 안전하고 효율적인 운전 결정을 내린다. 이러한 진보는 자동차의 안전성과 효율성을 크게 향상시킨다.

자율주행 자동차 분야에서 DNN의 발전은 테슬라와 웨이모와 같은 기업들의 최신 기술과 연구를 통해 두드러지게 나타난다. 예를 들어, 테슬라는 자율주행 시스템을 위한 전체적인 AI 기술을 통합하고 있으며, 테슬라의 Full Self-Driving (FSD) v12[2]는 특히 “End-to-End AI”로 발전하고 있다. 이러한 시스템들은 하드웨어 자원이 제한적이기 때문에, 고성능 컴퓨터에서 사용되는 DNN 모델을 그대로 적용하기 어렵다. 따라서 시스템에 적합한 모델의 효율적인 구현과 최적화가 중요한 과제로 부상하고 있다.

임베디드 및 실시간 시스템은 복잡한 요구사항과 제약 조건을 가지고 있다. 이들 시스템은 제한된 메모리 및 처리 능력, 엄격한 에너지 제약 조건 그리고 실시간 반응이 필수적인 환경에서 작동해야 한다. 예를 들어, 자율주행 차량은 이러한 시스템의 대표적인 예시로, 실시간 데이터 처리 및 신속한 의사결정이 필수적이다. 자율주행 차량에서 DNN은 차량의 센서 데이터를 분석하고, 실시간으로 환경 변화에 반응하여 안전한 운전 경로를 결정한다. 이러한 고도의 처리 능력과 신속한 반응은 DNN 구현에 있어 표준 모델의 단순한 적용을 넘어서는 도전을 제시한다. 이는 자율주행 차량의 안전성과 효율성에 직접적인 영향을 미치며, 이 분야에서 DNN 기술의 중요성을 강조한다.

이에 따라, 임베디드 및 실시간 시스템을 위한 DNN 추론 최적화에 관련된 연구가 활발히 진행되고 있다. 이러한 연구들은 모델의 경량화, 추론 지연 시간 최소화, 메모리 사용량 최소화, 에너지 소비 최소화 등 다양한 측면에서 이루어지며, 제한된 자원을 효율적으로 활용하는 방법에 초점을 맞추고 있다.

본 논문에서는 이러한 최적화 기법들의 최근 동향을 살펴보고, 임베디드 및 실시간 시스템에서의 DNN 활용을 위한 전략적 방향성에 대해 논의하고자 한다. 이 분야의 혁신적 접근과 기술적 발전은 제한된 자원을 가진 시스템에서도 DNN을 다양한 방식으로 효율적으로 작동시킬 수 있게 한다. 일부 연구는 추론 지연 시간을 줄이기 위해 새로운 DNN 아키텍처와 알고리즘을 개발하는 데 초점을 맞추고 있으며, 다른 연구들은 모델 경량화 및 효율적인 데이터 처리 방식을 통해 메모리 요구량을 줄이고 에너지 효율성을 개선하는 방향으로 진행되고 있다.

논문은 네 부분으로 구성된다. 먼저, 서론에 이어 딥 뉴럴 네트워크와 임베디드 및 실시간 시스템의 개념을 설명한다. 그 후, 이러한 시스템에서 DNN을 최적화한 최근의 연구 동향을 소개한다. 마지막으로, 결론과 향후 연구 전망을 제시하며, 이는 DNN 최적화와 관련된 기술적 과제 및 해결 방안을 탐색하는 데 중점을 두고 있다.

2. 딥 뉴럴 네트워크와 임베디드 및 실시간 시스템

2.1 딥 뉴럴 네트워크

딥 뉴럴 네트워크(DNN)는 인간 뇌의 신경망을 모

* 종신회원

방하여 구축된 인공 신경망의 일종이다. 이 기술은 복잡한 데이터 패턴을 학습하고, 이를 통해 분류, 예측, 인식 등의 다양한 작업을 수행한다. DNN은 인공 지능 및 기계 학습 분야에서 중요한 발전을 가져오며, 데이터에서 복잡한 관계와 패턴을 이해하고 해석하는 데 핵심적인 역할을 한다.

DNN은 대규모 데이터셋을 사용하여 패턴을 학습한다. 네트워크는 매개변수(가중치와 편향)를 조정하여 데이터의 특징을 효과적으로 추출하고 분석한다. 이러한 학습 방식은 네트워크가 복잡한 데이터 구조를 이해하고, 이를 바탕으로 정확한 예측과 결정을 내리는 데 기여한다.

DNN의 구조는 다양한 유형의 계층으로 구성되며, 이들은 데이터 처리 과정에서 각기 다른 역할을 수행한다. 기본적으로 입력 계층(input layer), 여러 숨겨진 계층(hidden layers), 그리고 출력 계층(output layer)을 포함한다. 입력 계층은 원시 데이터를 받아들이고, 숨겨진 계층들은 데이터를 처리한다. 데이터는 각 계층을 통과하며 더 높은 수준의 추상화를 거친다. 데이터는 [그림 1]과 같이 순전파를 통해 네트워크를 통과하고, 출력 계층에서 최종 결과가 생성된다.

DNN을 구성하는 계층은 학습 계층과 비학습 계층으로 분류된다. 학습 계층에는 컨볼루션 계층(Convolutional Layer)과 완전 연결 계층(Fully Connected Layer)이 포함된다. 컨볼루션 계층은 이미지와 같은 데이터에서 중요한 특징을 추출하며, 완전 연결 계층은 분류나 예측을 수행한다. 비학습 계층에는 풀링 계층(Pooling Layer), 정규화 계층(Normalization Layer), 드롭아웃 계층(Dropout Layer) 등이 포함된다. 풀링 계층은 입력 데이터의 크기를 줄이며, 정규화 계층은 데이터를 정규화하여 네트워크의 학습 안정성을 높이고, 드롭아웃 계층은 과적합을 방지한다. 이러한 비학습 계층들은 DNN의 효율성과 성능을 최적화하는 데 중요하다.

DNN 추론에서 종단 간 개념은 데이터 처리의 전 과정을 아우른다. 이는 카메라로 캡처한 이미지와 같은 원시 입력에서 시작하여 객체 인식과 같은 복잡한 작업을 수행하고, 최종 결과를 도출하는 과정을 포함한다. 이 과정은 자동화되어 있어, DNN이 복잡한 데이터를 효과적으로 처리하고 실시간으로 결정을 내릴 수 있게 해준다. 이러한 기능은 다양한 산업 분야에서 DNN의 응용 가능성을 확대하고, 실용적인 적용을 촉진한다.

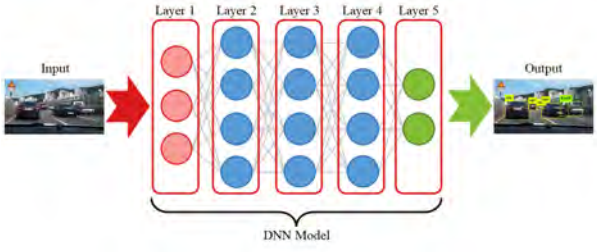

그림 1 딥 뉴럴 네트워크의 순전파

2.2 임베디드 및 실시간 시스템에서의 딥 뉴럴 네트워크

임베디드 시스템은 제한된 처리 능력과 메모리, 낮은 전력 소비가 요구되는 환경에서 작동하는 특수 목적의 컴퓨터 시스템이다. 이러한 시스템의 예로는 자동차의 제어 시스템, 휴대폰, 웨어러블 기기 등이 있다. 이 시스템에서 DNN의 성능은 메모리 구조와 제한된 자원에 의존하며, 이는 DNN 모델 최적화의 중요성을 강조한다. 모델의 크기나 연산의 복잡성을 줄이는 것은 제한된 자원 내에서도 효과적인 성능을 보장하는 데 필수적이다.

실시간 시스템은 엄격한 시간 제한 하에 작동하며, 주된 목표는 프로그램의 실행 시간을 시간 제한에 만족시키는 것이다. 이러한 시스템에서 최악 실행 시간(Worst Case Execution Time)과 최선 실행 시간(Best Case Execution Time)은 중요한 고려대상이다. 임베디드 시스템의 제한된 자원으로 인해 DNN의 추론 시간이 길어지면 실시간성을 보장하기 힘든 문제가 발생할 수 있으며, 이는 최적화를 통해 해결해야 한다.

모델 압축은 DNN 최적화의 가장 일반적인 접근 방법으로, [그림 2]에 표현된 프루닝과 양자화 기법을 포함한다[3][4][5]. 이러한 기법들은 모델의 크기를 줄이고 추론 속도를 향상시키지만, 모델의 정확도 감소라는 문제가 있다. 따라서, DNN의 장점을 유지하면서도 추론 지연 시간을 감소시키고, 자원 사용을 최적화하는 최적화 전략이 요구된다. 이러한 전략은 임베디드

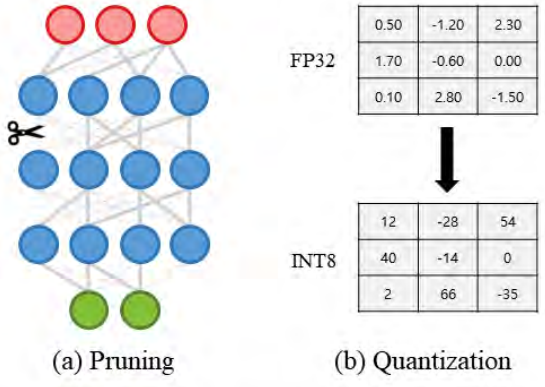

그림 2 딥 뉴럴 네트워크 모델 압축 기법

및 실시간 시스템에서 DNN의 활용 가능성을 확장하고, 시스템의 효율성과 신뢰성을 높이는 데 중요한 역할을 한다.

3. 최신 연구 동향 소개

임베디드 및 실시간 시스템에서 딥 뉴럴 네트워크(DNN)를 최적화하는 것은 여러 측면을 고려해야 한다. 이에는 DNN 추론 지연 시간 최소화, 메모리 요구량 최소화, 에너지 사용량 최소화, 자원의 활용성 극대화를 통한 성능 극대화, DNN 가용성 및 신뢰성 향상, 그리고 유연성 및 확장성 확보 등이 포함된다. 본 논문에서는 (1) 추론 지연 시간 최적화, (2) 메모리 요구량 최소화, 그리고 (3) 효율적인 자원 활용을 통한 성능 극대화 측면에서 DNN 최적화에 대한 최신 연구 동향을 소개할 예정이다.

3.1 추론 지연 시간 최적화

R-TOD[6]는 자율주행 차량의 안전을 위해 실시간 객체 탐지 시스템의 종단 간 지연을 분석하고 최소화하는 프레임워크이다. 이 연구는 객체가 탐지될 때까지의 [그림 3]에 표현된 종단 간 지연을 면밀히 조사하여 최적 및 최악의 상황에서의 지연을 예측하고, 이를 최소화하기 위한 세 가지 최적화 방법을 제시한다: (i) 수요에 따른 캡처(on-demand capture), (ii) 제로-슬랙 파이프라인(zero-slack pipeline), 그리고 (iii) 경쟁 없는 파이프라인(contention-free pipeline). 실험 결과, 이 최적화 방법들은 Darknet YOLO v3의 종단 간 지연을 1070ms에서 261ms로 76% 감소시켰다. 이 접근 방식은 신경망 구조 자체를 변경하지 않고 시스템 아키텍처만 수정하기 때문에 정확도에 영향을 주지 않는다. 이러한 연구는 자율주행을 위한 종단 간 지연 분석의 활용 가능성을 입증한다.

자율주행 자동차와 같은 실시간 객체 탐지 시스템에서는 다수의 카메라로부터 획득한 이미지를 처리할 때 정확성과 시간적 제약 문제가 발생할 수 있다. 이를 해결하기 위해 DNN-SAM[7]이라는 동적 분할 및 병합 DNN 실행 스케줄링을 가능하게 하는 프레임워크가 개발되었다. DNN-SAM은 전체 입력 이미지를 한 번에 처리하는 대신, DNN 추론 작업을 두 개의 작은 하위 작업으로 나눈다. 하나는 안전에 중요한 이미지 부분을 위한 필수 하위 작업이고, 다른 하나는 축소된 이미지를 처리하는 선택적 하위 작업이다. 이들은 독립적으로 실행되어 결과를 합친다. DNN-SAM은 중요도에 따라 하위 작업의 우선순위를 지정하고, 입력 이미지의 크기를 시간 제약에 맞춰 조정하여 필수 하위 작업의 응답 시간을 최소화하거나 선택적 하위 작업의 정확도를 극대화한다. 이 방법은 안전-중요 지역의 탐지 정확도를 2.0-3.7배 향상시키고 평균 추론 지연 시간을 4.8-9.7배 줄인다.

3.2 메모리 요구량 최소화

Demand Layering[8]은 DNN 추론을 실행할 때 GPU 메모리 부담을 줄일 수 있는 새로운 기법을 제시한 프레임워크이다. 기존에는 [그림 4a]처럼 DNN 모델의 매개변수가 실행 전 GPU 메모리에 로드되었으나, 이 방법은 특히 통합 GPU를 사용하는 임베디드 시스템에서 상당한 메모리 부담을 야기한다. Demand Layering은 고속의 고체 상태 드라이브(SSD)를 GPU의 협력 파트너로 사용하고, [그림 4b]처럼 DNN을 계층별로 실행하여 메모리 사용을 단일 계층 수준으로 최소화한다. 또한, 파이프라인 아키텍처를 통해 계층 실행과 매개변수 로딩 사이에 발생하는 추가 지연을 대부분 숨긴다. 이 방법은 대표적인 DNN 모델에서 메모리 사용량을 평균 96.5% 감소시키면서 지연 시간을 평균 14.8%만 증가시켰다. 또한, 메모리-지연 트레이드오프를 활용하여 지연 시간을 거의 0(1ms 미만)으로 줄이면서도 메모리 사용량을 88.4% 감소시킬 수 있다.

MCUNetV2[9]는 제한된 메모리 크기로 인해 MCU에서의 작은 규모의 딥 러닝 구현이 어려운 문제를 해결하는 네트워크다. 이는 CNN 설계에서 발생하는 메모리 분포의 불균형을 해결하기 위해, 패치별 추론 스케줄링을 제안하여 피쳐 맵의 작은 공간적 영역만을 작동시키고 최고 메모리 사용량을 크게 줄인다. 중복되는

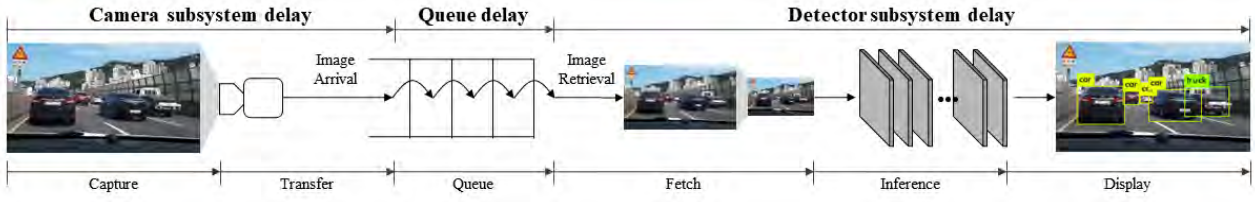

그림 3 실시간 객체 탐지 시스템의 종단 간 지연

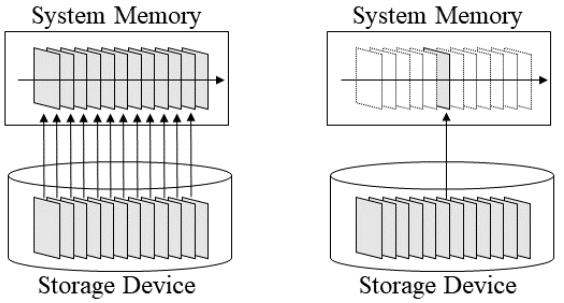

그림 4 파라미터 로딩 기법

패치와 계산 오버헤드를 줄이기 위해 접수 필드 재분배를 통해 나중 단계로 접수 필드와 FLOPs(FLoating point Operations Per Second)를 이동시키고 계산 오버헤드를 줄인다. MCUNetV2은 신경 구조 검색으로 신경 구조와 추론 스케줄을 함께 최적화하여 개발되었다. 패치 기반 추론은 기존 네트워크의 최고 메모리 사용량을 4-8배 감소시키며, MCU에서 ImageNet 정확도 기록(71.8%)을 세우고 32kB SRAM에서 시각적 단어 인식 데이터셋에서 90% 이상의 정확도를 달성했다. 또한, Pascal VOC에서 최신 결과보다 16.9% 높은 mAP를 달성함으로써 소형 디바이스에서의 객체 탐지를 가능하게 했다.

3.3 효율적인 자원 활용을 통한 성능 극대화

최근 발표된 연구[10]는 실시간 인지 시스템이 주어진 컴퓨팅 자원을 효율적으로 활용하여 시스템 성능을 극대화해야 한다는 점을 강조한다. 많은 인지 시스템들이 멀티스레드 파이프라인 구조를 채택하고 있지만, 이 구조는 불균형한 파이프라인 단계로 인해 중요한 파이프라인 정체를 발생시킨다. 또한, 제한된 작업 수준의 병렬성으로 인해 여러 CPU 코어를 완전히 활용하지 못하는 문제가 있다. 이를 해결하기 위해, 이 연구에서는 작업 병렬 아키텍처에서 데이터 병렬 아키텍처로의 전환을 제안한다. 이 아키텍처에서는 파이프라인 단계를 각 CPU 코어에 할당하는 대신 순차적 센서 데이터 도착을 모든 CPU 코어에 라운드 로빈 방식으로 분배한다. 이 시간적 병렬성을 활용함으로써, 인지 아키텍처는 CPU 코어 수에 비례하여 프레임 속도를 달성하며 거의 최적의 인지 지연을 제공한다. 전체 GPU 가속 대신, 선택적으로 DNN의 일부만을 가속하는 부분 GPU 가속을 적용하여, 프레임 속도 최적화와 지연 최적화를 모두 제공하는 유연한 시스템 구성을 제공한다.

LaLaRAND[11]는 실시간 임베디드 시스템에서 딥 뉴럴 네트워크(DNN) 실행을 가능하게 하는 스케줄링 기반 프레임워크이다. 현대의 기계 학습(ML) 프레임워크는 대부분 CPU와 GPU와 같은 이질적인 컴퓨팅 자원을 충분히 활용하지 못하고, 자원 할당의 정밀도가 낮고 DNN의 CPU와 GPU 실행 간의 비대칭성, 스케줄 가능성에 대한 인식이 부족하다. 이러한 문제를 해결하기 위해 LaLaRAND는 CPU 친화적인 양자화와 세밀한 CPU/GPU 자원 할당 체계를 통합하여 각 DNN 계층별로 유연한 CPU/GPU 스케줄링을 가능하게 한다. 이는 타이밍 보장을 손상시키지 않으면서 정확도 손실을 최소화한다. LaLaRAND의 구현은 기존 방법과 바닐라(PyTorch) 베이스라인에 비해 각각 56% 및 80% 더 많은 DNN 작업 세트를 스케줄할 수 있음을 보여주며, 성능(추론 정확도) 차이는 최대 -0.4%에 불과하다.

4. 결 론

본 논문은 임베디드 및 실시간 시스템에서의 딥 뉴럴 네트워크(DNN) 최적화에 대한 연구를 다루며, DNN의 구조적 혁신, 추론 지연 시간 감소, 메모리 및 에너지 요구량 최소화와 같은 여러 최신 동향을 탐구했다. 이러한 연구들은 제한된 자원을 가진 임베디드 시스템에서도 DNN의 효율적인 활용을 가능하게 함으로써, 자율주행 자동차, 의료 장비, 모바일 기기 등 다양한 분야에서의 응용을 촉진한다. 앞으로의 연구는 DNN 모델의 더욱 깊은 경량화, 실시간 처리 능력 향상, 그리고 에너지 효율성 증대에 초점을 맞출 것으로 예상된다. 특히 자율주행 자동차 분야에서 이 연구의 진보는 자동차의 안전성과 신뢰성을 높이는 데 중요한 역할을 할 것이며, 미래 자율주행 기술 발전에도 크게 기여할 것으로 기대된다. DNN 최적화 기술의 지속적인 발전은 자율주행 자동차를 포함한 여러 분야에서 혁신을 가져올 것으로 예상된다.

참고문헌

[1] J. Gorospe, R. Mulero, O. Arbelaitz, J. Muguerza, and M. Á. Antón, “A generalization performance study using deep learning networks in embedded systems,” Sensors, 2021, vol. 21, no. 4, p. 1031.

[2] “Full Self-Driving (Beta)”, https://www.tesla.com/support/recall-fsd-beta-driving-operations, 2023.

[3] S. Han, H. Mao, and W. J. Dally, “Deep Compression:

Compressing deep neural network with pruning, trained quantization and huffman coding," in Proc. 4th International Conference on Learning Representations (ICLR), 2016.

[4] Y. He, J. Lin, Z. Liu, H. Wang, L.-J. Li, and S. Han, "AMC: AutoML for model compression and acceleration on mobile devices," in Proc. 15th European conference on computer vision (ECCV), 2018, pp. 784-800.

[5] Z. Liu, J. Li, Z. Shen, G. Huang, S. Yan, and C. Zhang, "Learning efficient convolutional networks through network slimming," in Proc. 16th IEEE international conference on computer vision (ICCV), 2017, pp. 2736-2744.

[6] W. Jang, H. Jeong, K. Kang, N. Dutt, and J.-C. Kim, "R-tod: Real-time object detector with minimized end-to-end delay for autonomous driving," in Proc. 41st IEEE Real-Time Systems Symposium (RTSS), 2020, pp. 191-204.

[7] W. Kang, S. Chung, J. Y. Kim, Y. Lee, K. Lee, J. Lee, K. G. Shin, and H. S. Chwa, "DNN-SAM: Split-and-merge dnn execution for realtime object detection," in Proc. 28th IEEE Real Time Technology and Applications Symposium (RTAS), 2022, pp. 160-172.

[8] M. Ji, S. Yi, C. Koo, S. Ahn, D. Seo, N. D. Dutt, and J.-C. Kim, "Demand layering for real-time DNN inference with minimized memory usage," in Proc. 43rd IEEE Real-Time Systems Symposium (RTSS), 2022, pp. 291-304.

[9] J. Lin, W.-M. Chen, H. Cai, C. Gan, and S. Han, "MCUNetV2: MemoryEfficient Patch-based Inference for Tiny Deep Learning," in Proc. 35th Neural Information Processing Systems (NeurIPS), 2021, pp. 1-13.

[10] S. Ahn, S. Kim, H. Kang, and J.-C. Kim, "Data-parallel Real-Time Perception System with Partial GPU Acceleration for Autonomous Driving," in Proc. 8th Machine Learning for Autonomous Driving Symposium (ML4AD), 2023.

[11] W. Kang, K. Lee, J. Lee, I. Shin, and H. S. Chwa, "Lalarand: Flexible layer-by-layer CPU/GPU scheduling for real-time DNN tasks," in 42nd IEEE Real-Time Systems Symposium (RTSS), 2021, pp. 329-341.

약 력

안 솔

2022 국민대학교 자동차IT융합학과 졸업(학사)
2022~현재 국민대학교 자동차공학전문대학원 자동차IT융합전공 석박사통합과정
관심분야: Real-Time System, Real-Time DNN Inference, DNN Optimization
Email : solahn00@kookmin.ac.kr

김 종 찬

1999 서울대학교 전산학과 졸업(학사)
2001 서울대학교 전기컴퓨터공학부 졸업 (석사)
2013 서울대학교 전기컴퓨터공학부 졸업 (박사)
2002~2008 티맥스소프트 연구원
2014~현재 국민대학교 자동차IT융합학과 교수
관심분야 : Real-Time System, Real-Time DNN Inference, Truck Platooning
Email : jongchank@kookmin.ac.kr