

# 생산 현장에서 가상화 리소스 할당을 반영한 효율적인 AIoT 리소스 관리 기법

정운수\*, 김용태\*\*

## Efficient AIoT Resource Management Techniques Reflecting Virtualization Resource Allocation in Production Environments

Yoon-Su Jeong\*, Yong-Tae Kim\*\*

---

이 논문은 2022학년도 한남대학교 학술연구비 지원에 의하여 연구되었음

---

### 요약

인공지능과 사물인터넷의 융합이 가속화되면서 오늘날 많은 기업들이 경쟁력과 효율성을 높이기 위해 산업 현장에서 AIoT를 광범위하게 사용하고 있다. 산업 현장에서 컴퓨팅 집약적이고 리소스 집약적인 네트워크 에지는 효율적이지만, 리소스 용량 및 전력 예산에는 낮은 대기 시간과 에너지 효율성을 높일 수 있는 관리 방법이 필요하다. 본 논문에서는 AIoT 자원 관리의 모델 성능(정확성 및 견고성)과 자원 비용을 동시에 최적화할 수 있는 가상화된 자원 할당 기반 AIoT 자원 관리 기법을 제안한다. 제안 방법은 라푸노프(Lyapunov) 최적화 이론을 사용하여 에지 및 클라우드 리소스의 병목 현상을 최소화하여 저비용으로 지속적인 학습을 수행한다. 또한, 제안 방법은 AIoT 리소스의 지속적인 학습 모델에 가중치를 지속적으로 업데이트하여 가상화 리소스 할당을 통해 리소스의 처리 지연 및 네트워크 오버헤드를 최소화한다.

### Abstract

As the convergence of artificial intelligence and the Internet of Things accelerates, many companies today use AIoT extensively in their industrial sites to increase competitiveness and efficiency. In industrial areas, computing-intensive and resource-intensive network edges are efficient, but resource capacity and power budgets require management methods that can increase low latency and energy efficiency. This paper proposes a virtualized resource allocation-based AIoT resource management technique that can simultaneously optimize the model performance(accuracy and robustness) and resource cost of AIoT resource management. The proposed method uses Lyapunov optimization theory to minimize bottlenecks in edge and cloud resources to perform continuous learning at a low cost-efficiently. In addition, the proposed technique continuously updates weights in the constant learning model of AIoT resources to minimize processing delays and network overhead of resources through virtualization resource allocation.

### Keywords

virtual resource, AIoT, Lyapunov optimization theory, resource allocation, virtualization, resource management

---

\* 목원대학교 게임소프트웨어공학과 교수

- ORCID: <https://orcid.org/0000-0003-3455-5947>

\*\* 한남대학교 멀티미디어공학과 교수(교신저자)

- ORCID: <https://orcid.org/0000-0002-1554-1040>

• Received: Jun. 05, 2023, Revised: Jun. 28, 2023, Accepted: Jul. 01, 2023

• Corresponding Author: Yong-Tae Kim

Dept. of Multimedia Engineering, Hannam University, Korea

Tel.: +82-42-829-7678, Email: [ky7762@hnu.kr](mailto:ky7762@hnu.kr)

## 1. 서 론

최근 대기업들은 인공지능과 IoT 기술을 결합한 생산 현장을 대규모 단지로 건설하고 있다[1]. 그러나, 생산 현장의 IoT 장치에서 수집된 많은 양의 정보를 효율적으로 처리하려면 네트워크 에지가 컴퓨터 집약적이어야 하며 IoT 리소스를 대량으로 처리할 수 있어야 한다.

가상화 리소스 할당을 반영한 효율적인 AIoT(인공지능 사물인터넷) 관리 기법은 가상화 기술, 자원 관리 및 스케줄링, 동적 리소스 조정, 에지 컴퓨팅 활용, 보안과 개인정보보호 등 다양한 측면에서 고려된다.

AIoT 환경에서 가상화 기술은 가상 리소스를 생성하고 관리할 수 있다. 가상화 기술은 컴퓨팅, 스토리지 및 네트워킹을 포함한 다양한 리소스를 가상화하여 효율적인 리소스 할당을 지원한다. 가상화를 통해 필요한 리소스를 동적으로 조정하고 서버 또는 장치 이중화를 구현하여 안정성과 가용성을 높일 수 있다.

가상화된 리소스를 효율적으로 관리하고 할당하려면 리소스 관리 및 스케줄링 기술이 필요하다. 이를 위해서는 AIoT 시스템의 요구사항을 분석하고 자원의 가용성과 성능을 고려하여 최적의 할당 계획을 결정해야 한다. 예를 들어, 머신러닝 알고리즘은 실시간 데이터를 분석하고 예측하여 자원 할당을 최적화할 수 있다.

AIoT 환경에서는 변화하는 데이터 볼륨 및 처리 요구사항을 실시간으로 충족할 수 있도록 AIoT 리소스를 동적으로 조정해야 한다. AIoT는 가상화된 리소스를 사용하여 리소스 할당을 자동으로 조정하고 까다로운 장치에 대한 처리 우선순위를 지정할 수 있다. 이 처리는 시스템 성능을 최적화하고 유연한 운영을 가능하게 한다.

AIoT 환경에서는 빅 데이터 처리 및 분석을 위해 많은 양의 데이터가 필요하다. 이를 위해 에지 컴퓨팅과 클라우드 컴퓨팅을 활용할 수 있습니다. 에지 컴퓨팅은 데이터를 처리하고 장치 주변을 분석하여 대역폭 소비와 대기 시간을 줄인다. NAT은 에지 장치에서 가상화된 리소스를 효과적으로 사용하고 중앙 집중식 리소스와 협력하여 최적의 처리

환경을 구축한다.

AIoT 환경에서는 다양한 장치와 시스템이 상호 연결되므로 보안 및 개인 정보 보호에 대한 고려가 필요하다. 가상화 리소스 할당을 반영하는 AIoT 관리 기법에는 데이터 암호화, 액세스 제어 및 인증을 포함한 다양한 보안 메커니즘이 필요하다. 또한, 개인 정보 보호를 위해 데이터 수집, 저장 및 전송 규정을 준수해야 한다.

이러한 기술을 통해 AIoT 환경에서 가상화된 리소스 할당을 효율적으로 관리할 수 있다. 그러나 AIoT 시스템의 특성과 요구사항에 따라 최적의 방법을 선택하고 결합하는 것이 필수적이다.

클라우드 환경으로 생산 현장을 구축하는 경우 IoT 리소스 용량 및 전력 버짓을 고려하여 낮은 대기 시간과 에너지 효율을 높일 수 있는 관리 방법이 필요하다[2]. 최근 연구에서는 전력 효율성, 지연 및 계산 속도와 같은 성능 목표를 최적화하는 것을 목표로 많이 연구되고 있다[3][4]. T. X. Tran et al.은 오프로딩 과정에서 사용자의 작업을 극대화하기 위해 공동 작업 오프로딩 및 리소스 할당 문제를 최적화하는 방법을 제안하였다[3]. L. Zhao et al.은 평균 서비스 응답 시간 최소화를 위해 에지 리소스를 할당하는 방법을 조사했다[4].

그러나 IoT 장치의 행동 분석에 관한 기존 연구 대부분은 리소스 수요 측면에서 이기종 IoT 장치의 복잡한 행동을 모델링하기 위해 더 포괄적일 필요가 있다. 또한, 이기종 IoT 기기의 자원 수요 행동(자원 수요 행동)은 자원 사용량, 기간, 자원 요청 밀도 등 에지 컴퓨팅 서비스 정보를 고려하여 프로파일링해야 한다. 모델링 방법은 다양한 형식, 길이 및 시간 관계(예: 시계열 및 비 시계열 데이터)에서 BRD 데이터를 측정해야 한다.

본 논문에서는 생산 현장에서 IoT 자원 관리를 최적화하기 위한 가상화 자원 할당 기반의 AIoT 자원 관리 기법을 제안한다. 제안 방법은 라푸노프(Lyapunov) 최적화 이론을 활용하여 모델 성능(정확성 및 견고성)과 AIoT 자원 관리의 자원 비용을 동시에 최적화한다. 제안 방법은 저렴한 비용으로 효율적인 연속 학습을 수행할 수 있어서 가상화 리소스 할당을 통해 에지 및 클라우드 리소스의 병목 현상, 처리 리소스의 지연, 네트워크 오버헤드를 최소화한다.

IoT 자원의 관리 비용을 최적화하기 위한 제안 방법의 목적은 다음과 같습니다. 첫째, 에지 DNN 모델을 사용하여 IoT 리소스를 학습하기 위해 모델 가중치를 지속적으로 업데이트한다. 둘째, 각 슬롯에 알려진 새로운 AIoT 리소스 샘플은 두 부분(에지, 클라우드)에 적용되어 DNN 모델에서 병렬로 재교육할 수 있다. 셋째, 클라우드와 에지 자체가 획득한 AIoT 리소스는 현재 시간 슬롯  $t$ 에서 새로운 데이터 샘플에 해당하는 모델 가중치  $W(t)$ 를 추가 계산한다.

이 논문의 구성은 다음과 같다. 2장에서는 IoT 가상화 자원과 관련된 기존 연구를 분석한다. 3장에서는 가상화 자원 할당을 반영한 AIoT 자원 관리 기법을 제안하고, 4장에서는 성과 평가를 수행한다. 마지막으로 5장은 결론을 맺는다.

## II. 관련 연구

### 2.1 라푸노프 최적화 이론

Lyapunov 최적화 이론은 동적 시스템의 동작을 분석하고 최적화하는 데 사용되는 수학적 프레임워크이다. Lyapunov 최적화 이론은 동적 시스템의 안정성, 수렴 및 성능 특성을 연구하고 시스템 동작을 최적화하기 위한 제어 전략을 설계하기 위한 강력한 도구 세트를 제공한다.

Lyapunov 최적화 이론은 동적 시스템의 안정성을 연구하기 위해서 일련의 미분 방정식 또는 차분 방정식을 사용하여 동적 시스템의 각 상태에 스칼라 값을 할당하는 Lyapunov 함수를 정의한다. Lyapunov 함수는 시스템의 궤적을 따라 감소하도록 선택되어 안정적인 평형 또는 원하는 동작을 수렴하도록 시스템의 안정성 특성을 특성화하는 데 사용된다.

예를 들어, 상태 벡터  $x$ 로 설명되는 동적 시스템과 방정식  $dx/dt = f(x)$ (여기서  $f$ 는 벡터 값 함수임)로 주어진 시간에 따른 진화를 고려할 경우, Lyapunov 함수  $V(x)$ 는 다음과 같이 정의된다.

- $V(x)$ 는 양의 정부호이다. 모든  $x \neq 0$ 에 대해  $V(x) > 0$ 이고 관심 평형점에서만  $V(x) = 0$ 이다.
- $V(x)$ 는 반경 방향으로 무한하다:  $V(x) \rightarrow \infty$  as  $\|x\| \rightarrow \infty$

- $dV(x)/dt$ 로 표시되는 Lyapunov 함수의 시간 도함수는 시스템의 안정성을 분석하는 데 사용된다.
- $dV(x)/dt$ 의 부호는 시스템 동작에 대한 통찰력을 제공한다.
- 모든  $x \neq 0$ 에 대해  $dV(x)/dt < 0$ 이면 시스템은 전역적으로 점근적으로 안정하다. 즉, 모든 초기 조건에서 평형점으로 수렴한다.
- 모든  $x \neq 0$ 에 대해  $dV(x)/dt \leq 0$ 이면 시스템은 전체적으로 안정적이지만 반드시 수렴할 필요는 없다. 제한 주기 또는 진동 동작이 있을 수 있다.
- 모든  $x \neq 0$ 에 대해  $dV(x)/dt > 0$ 이면 시스템이 불안정하다.

Lyapunov 최적화 이론은 안정성 분석을 넘어 최적화까지 확장할 수 있다. Lyapunov 기능을 신중하게 선택하고 제어 입력을 도입함으로써 에너지 소비 최소화, 성능 최대화 또는 원하는 목표 달성과 같은 시스템 동작을 최적화하는 제어 전략을 설계할 수 있다.

### 2.2 기존 연구

F. Foroghifar et al.은 AIoT 자율 기능을 확장하기 위한 알고리즘의 복잡성과 신뢰성을 고려한 자기 인식 개념에 기반한 효율적인 접근 방식을 제안했다[5]. 이 접근 방식은 자기 인식 개념을 기계 학습 계산에 적용하여 복잡하고 소모적인 기계 학습 계산을 에지, 안개 및 클라우드 간에 분산시킨다.

Y. Kang et al.은 엔드 투 엔드 지연 시간과 모바일 에너지 소비를 줄이기 위해 모바일 장치와 데이터 센터 간에 DNN 계산을 분할하는 Neuroon 경량 스케줄러를 제안했다[6]. Z. Zhao et al.은 산업 IoT 네트워크에서 대기 시간과 에너지 소비를 줄이기 위해 대역폭 할당, 오프로드 및 릴레이 선택을 포함하는 모바일 에지 컴퓨팅(MEC)을 위한 새로운 3계층 최적화 프레임워크를 제안했다[7]. B. Yang et al.은 자원 제약이 있는 무인 항공기(UAV)를 위한 효율적인 계층적 DL(Deep Learning) 작업 오프로드 프레임워크를 제안했다[8]. 이 프레임워크에서 UAV는 일부 사전 훈련된 CNN 모델의 하위 계층에 내장됐지만 클라우드 서버는 CNN 모델의 상위 계층을 처리한다.

H. Gu et al.은 컨볼루션 샴(Convolutional Siamese) 네트워크에 기반한 객체 추적을 위한 새로운 아키텍처를 제안했다[9]. Liu et al.은 증분 학습과 낮은 데이터 전송으로 CNN 업데이트의 오버헤드를 줄이는 FitCNN이라는 클라우드 지원 DNN 프레임워크를 제안했다[10]. C. Gamanayake et al.은 필터와 기본 하드웨어 아키텍처의 중요성을 고려하여 클러스터 가지치기라는 새로운 접근법을 제안했다[11]. T. D. Nguyen et al.은 손상된 IoT 장치를 탐지하기 위해 장치 유형별 통신 프로필에 구축된 DioT라는 자율 연합 자체 학습 분산 시스템을 제안했다[12]. S. Wang et al.은 IoT 장치의 리소스 예산 손실을 최소화하기 위해 로컬 업데이트와 글로벌 매개변수 집계 간의 최적 균형을 계산하는 알고리즘을 제안했다[13]. J. Mills et al.은 분산 데이터 세트와 통신 비용을 고려하여 수렴 속도를 높이는 기술을 제안했다[14]. 이 기술은 런타임 감소를 위해 분산형 형태의 Adam 최적화를 사용했다. 그러나 보다 정확한 FL 추론을 위해 일부 기존 작업만 클라우드와 협업한다.

III. 가상화 리소스 할당을 반영한 AIoT 자원 관리 기법

3.1 개요

AIoT는 기업의 다양한 생산 환경에서 널리 사용되기 때문에 생산 환경에서 생산되는 대량의 IoT 리소스를 지속적으로 탐지해야 한다. 특히 기업의 생산 환경에서 에지 인텔리전스 모델(예: DNN 등)을 생성하도록 빅 데이터를 교육하면 AIoT 리소스 요구사항이 증가한다. 에지 컴퓨팅 노드의 용량을 크게 초과하므로 에지와 클라우드를 조정하여 에지 인텔리전스 모델을 교육하는 것이 적절하다.

제안된 기법은 에지와 클라우드가 별도의 타임슬롯 모델을 고려하여 가상화 리소스를 할당하기 때문에 AIoT 리소스에 대한 시간 지연 데이터의 도착/처리 시간 및 리소스 비용을 특성화해야 한다.

에지 컴퓨팅 환경에 도착하는 IoT 데이터는 일반적으로 엔드 장치의 이동성으로 인해 다양한 방식으로 예측되지 않는다. 제안 방법은 그림 1과 같이 에지 DNN 모델을 사용하여 IoT 자원의 지속적인 학습을 보장하기 위해 모델 가중치를 지속해서 업데이트하고, 새로운 데이터 샘플에 DNN 모델을 주기적으로 유지함으로써 모델의 정확도를 점진적으로 향상시킨다.

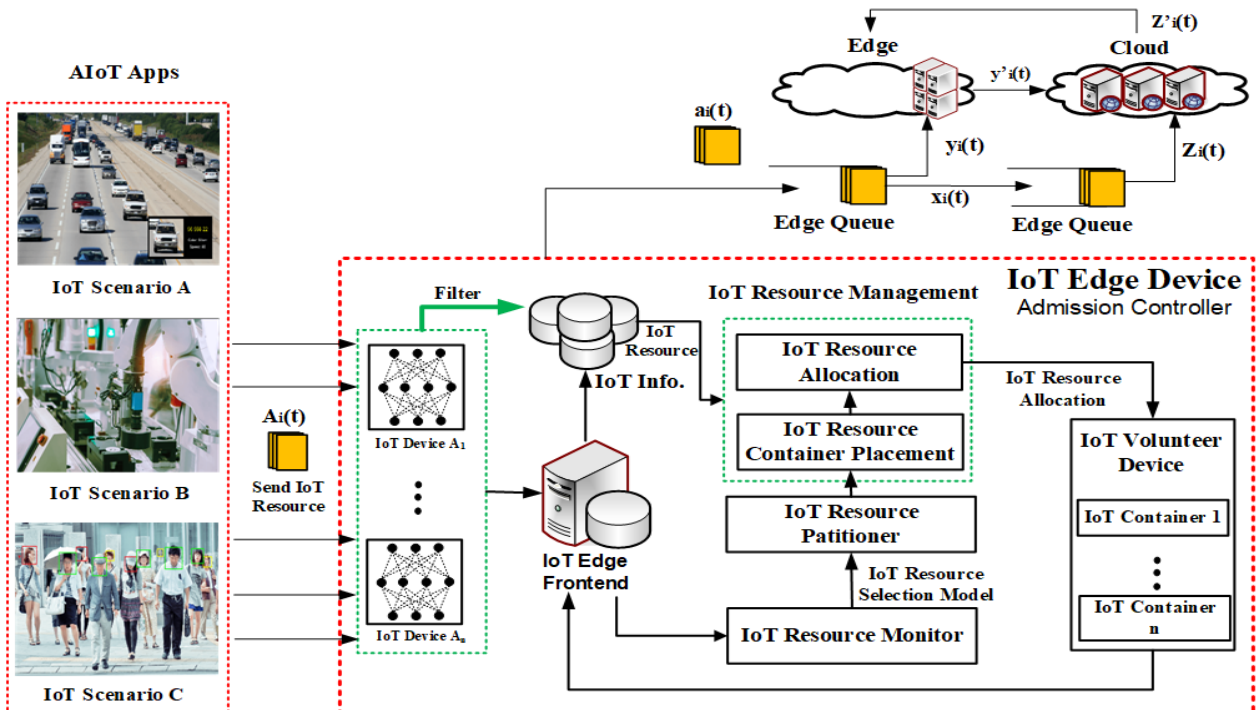


그림 1. 다중 AIoT 리소스의 지속적인 학습 모델  
 Fig. 1. Continuous learning model for multiple AIoT resources

또한, 제안 방법은 그림 2와 같이 클라우드 에지 서버의 시너지 효과를 지속적으로 반영할 경우 각 슬롯에서 학습된 새로운 리소스 샘플이 두 부분(에지, 클라우드)으로 분할되도록 전송된다. 이 두 부분은 DNN 모델에 대한 재교육을 병렬로 수행한다.

자원 샘플의 각 부분은 DNN 모델을 독립적으로 훈련시키고 시간이 지남에 따라 모델 가중치를 얻은 다음 모델 가중치를 클라우드에 전달한다. 모델 가중치  $W_e(t)$  및  $W_c(t)$ 는 AIoT 리소스 모델의 평균 연산(즉, 모델 집계)을 수행한다.

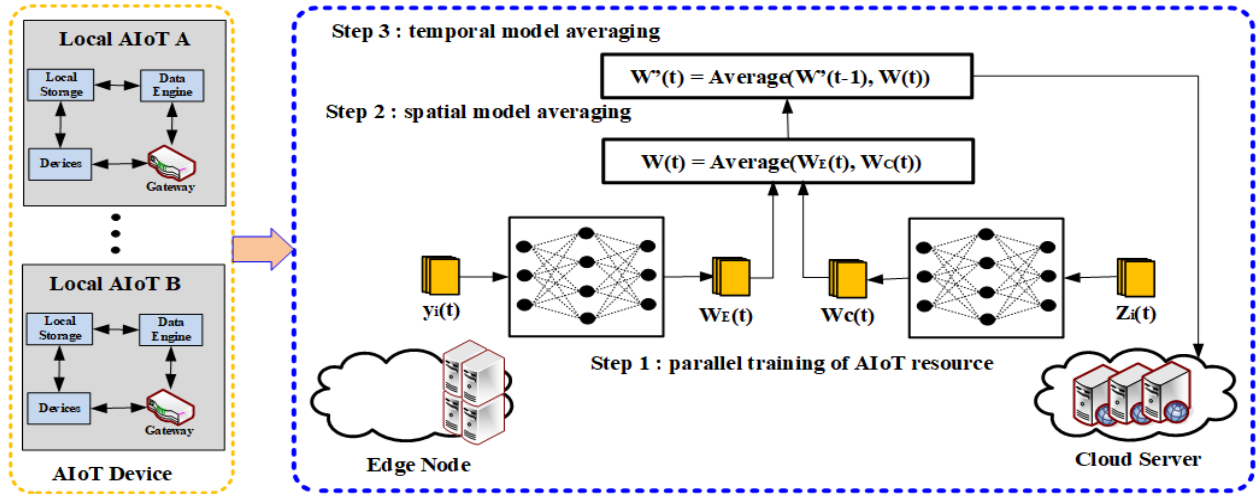


그림 2. AIoT 리소스의 지속적인 학습 단계 과정  
Fig. 2. Continuous learning phase course for AIoT resources

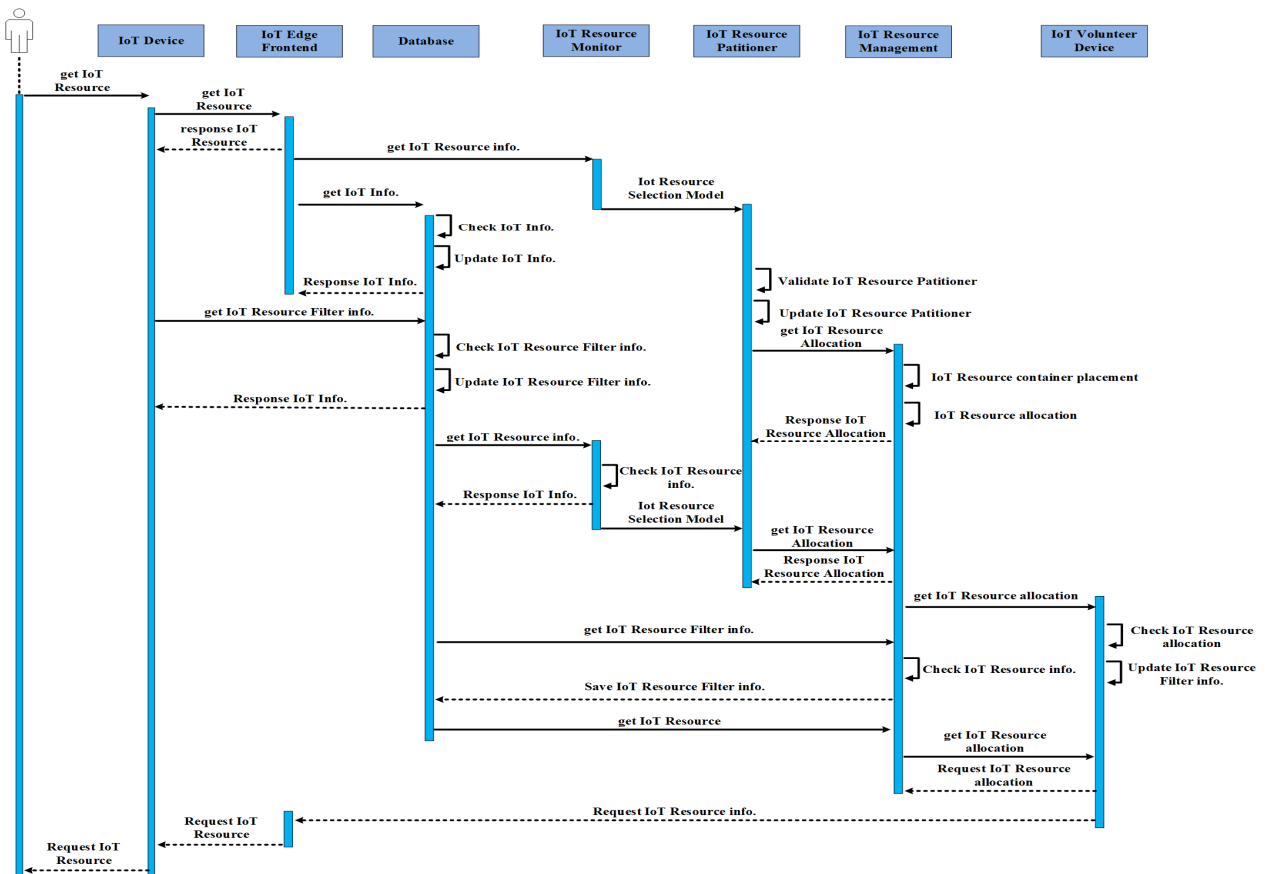


그림 3. AIoT 리소스의 다이어그램 동작 처리  
Fig. 3. Diagram process of AIoT resources

클라우드에서 획득한 예지는 현재 시간 슬롯  $t$ 에서 새 데이터 샘플에 해당하는 모델 가중치  $W(t)$ 를 계산한다. 현재 시간 슬롯의 모델 가중치  $W(t)$ 와 과거 시간 슬롯의 과거 모델 가중치  $W(t-1)$ 에 대해 추가 시간 모델 평균화 작업을 수행하여 예지는 최신 데이터 샘플의 지식을 통합하는 업데이트된 가중치  $W(t)$ 를 생성한다. 생성된  $W(t)$ 는 클라우드에 다시 입력되어 재교육 된다.

### 3.2 AIoT 리소스의 시퀀스 다이어그램

제안 방법은 IoT 기기에서 송·수신되는 IoT 자원을 에지 환경에서 안정적으로 처리하기 위해 그림 3과 같은 시퀀스 다이어그램을 통해 IoT 자원을 처리한다. IoT 자원을 처리하는 중에 에지 서버 및 클라우드로 전송되는 AIoT 리소스는 AI 알고리즘(머신러닝, 패턴 인식, 예측 분석 등)을 사용하여 처리 및 분석된다. 특히, IoT 에지 기기는 IoT 자원을 인식하고 IoT 자원봉사자(Volunteer)에서 처리할 수 있는 추가 AIoT 리소스 정보만을 추출하기 때문에 AIoT 리소스 손실을 최소화할 수 있다.

그림 3의 시퀀스 다이어그램은 클라우드 컴퓨팅과 결합되어 AIoT 기능의 요구사항을 충족하기 때문에 애플리케이션에서 생성된 AIoT 리소스를 클라우드 서버로 전달할 때는 네트워크 지연 시간이 개선될 수 있다. 그림 3에서 AIoT 기기는 IoT 리소스를 순차적으로 처리하기 때문에 시간이 지남에 따라 IoT 리소스가 일치되고 검증된다. 제안 방법은 AIoT가 IoT 리소스를 중앙에서 운영하기 때문에 IoT 수와 관계없이 IoT 리소스 간 연결을 자동화한다. 제안 방법은 IoT 리소스의 안정성을 보장하기 위해 클라우드 서비스 요구사항에 따른 정책에 의해 지원된다.

### 3.3 AIoT 리소스의 알고리즘

제안 기법은 AIoT 리소스들을 생성한 후 에지 큐와 클라우드 큐의 리소스를 갱신하기 위해서 표 1과 같은 알고리즘을 AIoT 리소스 관리자가 사용한다. 표 1에서 AIoT 리소스들은 임의의 크기로 그룹

화한 후 AIoT 리소스 정책과 함께 저장된다. AIoT 리소스는 이전 패턴의 AIoT 리소스와 해시값으로 저장한 후 연결되기 때문에 AIoT 리소스를 효과적으로 검증할 수 있다. AIoT 대기열에 따라 공동 승인 제어, 전송 스케줄링 및 자원 프로비저닝을 위한 최적값을 반영하여 각 시간 슬롯을 업데이트한다.

제안 방법에서는 AIoT 리소스의 시퀀스 번호와 홀수/짝수 복제 횟수에 따라 해시값이 구성된다. 해시값을 통해 인접 AIoT 리소스를 손실 없이 지속적으로 처리할 수 있어 AIoT 리소스 생성을 최적화하고 검증 오버헤드를 줄일 수 있다.

표 1. AIoT 리소스 처리 알고리즘  
Table 1. AIoT resource process algorithm

Input : AIoT resources collected from IoT devices Output : Updating resources in edge queues resource and cloud queues resource
Initialize the control parameter, edge queue, cloud queue for each time slot do if Receive AIoT resource from IoT devices then Check the current edge queue Compare AIoT resource received from other AIoT resource with resource database Store and update the AIoT resource else Request AIoT resource received from other AIoT resource end if for each application do if edge queue > control parameter then let admission control = 0; joint transmission scheduling, edge resource provisioning and cloud resource provisioning else let admission control = cloud queue; end if Update the edge queue and the cloud queue end for Convert each block to replication if Generate hash values for odd/even resource Add to first and last of multiple hash chains Verification of integrity of IoT Edge resource else Regenerate hash values for odd/even end if end for

### 3.4 블록체인 기반 IoT 리소스 연결

제안 방법은 AIoT 리소스가 연결될 때 각 AIoT 리소스에 대한 특정 확률에 가중치를 부여한다. 가중치가 부여된 AIoT 리소스는 그림 4와 같이 AIoT 리소스 속성에 따라 그룹화 프로세스를 주기적으로 수행한다. 그림 4는 AIoT 리소스 중 서로 연결된 AIoT 리소스 간의 해시 체인 연결 과정을 보여준다. AIoT 리소스가 실시간으로 처리되어야 하므로 해시 체인은 지연된 처리 시간을 최소화하면서 AIoT 리소스의 무결성을 보장할 수 있도록 그룹화가 진행된다. 또한, 그림 4에서는 AIoT 리소스 블록에 장애가 발생할 경우 AIoT 리소스 블록 중 임의의 블록을 추정하여 AIoT 리소스의 무결성을 확인한다. 그림 4에서 AIoT 자원을 연결하는 동안 오류가 발생하면 AIoT 서버는 AIoT 리소스에 대한 오류 판단을 수행하고 AIoT 리소스를 다시 연결한다.

이 절에서는 클라우드 환경에서 동작하는 AIoT 리소스의 효율성, 정확성, 처리 시간 등을 평가한다.

### 4.1 환경설정

그림 5는 AIoT 리소스를 송·수신하는 데 사용되는 IoT 장치이다. 각 AIoT 서버는 다중 메시 네트워크로 구성된 것으로 가정한다. AIoT 서버는 각 에지 서버를 통해 AIoT 리소스를 실시간으로 정기적으로 수집하고 분석한다.

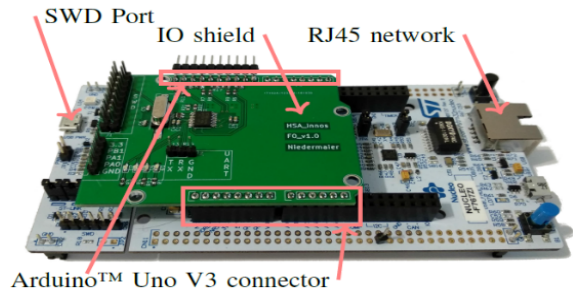


그림 5. 시뮬레이션을 위한 IoT 장치  
Fig. 5. IoT device for simulation

## IV. 평가

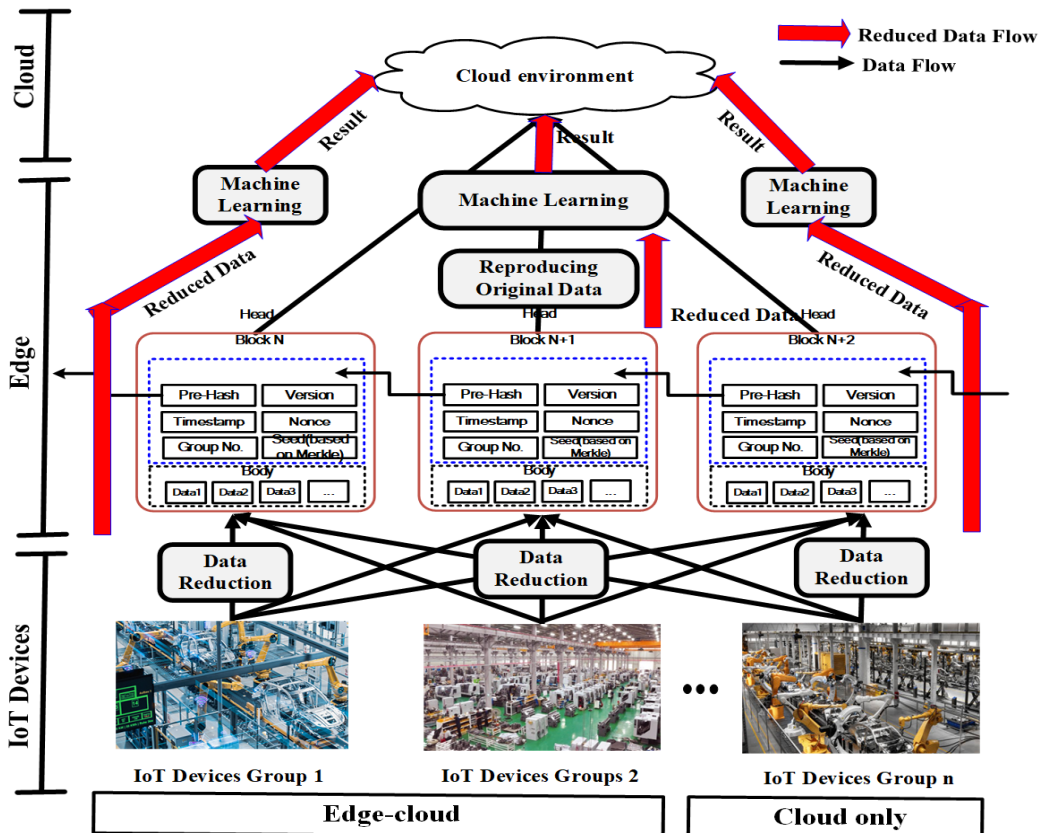


그림 4. AIoT 리소스 연결  
Fig. 4. Connection of AIoT resources

표 2는 성능평가에 사용되는 매개변수가 설정된 값을 보여준다. 시뮬레이션에서 IoT 장치를 포함한 네트워크 범위는 25m로 설정되며, 송·수신 대역폭은 5MHz/2.5MHz로 설정한다. IoT 장치의 정보 생성 시간은 5분마다 수행하였으며, 변동 보상 계수의 제한 시간은 0.01s/KB로 설정하였다. 기타 구성 값은 표 2와 같다.

표 2. 실험 파라미터 설정  
Table 2. Experimental parameter setup

Parameter	Value
The transmit/receive power	0.05W / 0.1W
The network coverage radius	25m
The static circuit power	0.01W
The available bandwidth	5MHz / 2.5MHz
The power of noise	-174dBm/Hz
Input data size	4 kbits/s
Delay threshold	5 s
Link capacity	10 Gbps
Poisson lambda	85%
Data generation span	5 min
The unit price of energy	0.1 Token/J

표 3은 AIoT 리소스를 모니터링하거나 분석하기 위해 IoT 리소스를 제공하여 피드백 정보를 반영할 수 있는 머신러닝 알고리즘(LR, Logistic Regression), MLP(Multi-layer Perceptron), SVM(Support Vector Machine), KNN(K-nearest Neighbor)의 시뮬레이션 매개변수의 설정값이다.

표 4. AIoT 리소스 수에 따른 성능 분석  
Table 4. Performance analysis by number of AIoT resource

Performance parameter	Number of IoT server															Total Average
	1					2					3					
	Number of IoT edge					Number of IoT edge					Number of IoT edge					
	1	2	3	4	Avg.	1	2	3	4	Avg.	1	2	3	4	Avg.	
Efficiency(%)	71.70	72.27	73.06	74.17	72.80	72.44	72.99	74.11	74.44	73.50	74.13	74.85	75.41	76.10	75.12	74.05
Accuracy(%)	65.25	65.72	66.06	66.63	65.91	65.96	66.34	66.84	67.07	66.55	66.47	66.78	67.29	67.81	67.09	66.57
Process time(ms)	2.59	2.84	3.14	3.88	3.12	2.77	3.06	3.48	4.34	3.41	2.93	3.33	3.79	4.82	3.72	3.48

표 3. 알고리즘별 파라미터값  
Table 3. Parameter values of algorithm

Algorithm	Parameter Value
LR	Regularization strength(C) : 10 Solver : L-BFGS penalty : L2
SVM	Regularization parameter(C) : 0.001, 0.01, 1, 10 Kernel : rbf Probability : true
KNN	Number of neighbors : 3~5 Weights : uniform

## 4.2 성능평가

AIoT 리소스의 효율성, 정확성, 처리 시간은 AIoT 내장 지능형 팜을 사용하여 표 4처럼 평가하였으며, DNN 서비스 모델은 Google Colab 지원 개발 언어 및 라이브러리를 사용하여 표 5처럼 분석하였다.

### 4.2.1 카테고리별 성능 평가

표 4는 AIoT 리소스를 서버에 전달할 때 에지 및 클라우드에서 AIoT 리소스의 성능 평가로 인한 효율성, 정확성 및 처리 시간을 보여준다. 표 4와 같이 AIoT 리소스를 AIoT 리소스 특성에 따라 주기적으로 그룹화하면서 실시간으로 처리하기 때문에 AIoT 기기 수 및 AIoT 기기 간 물리적 거리에 따라 AIoT를 사용하지 않을 때보다 평균 7.17%의 효율성이 향상되었다. AIoT 기기가 AIoT 리소스만 인식할 수 있도록 추가 정보만 추출하고 AIoT 자원 봉사자(Volunteer)가 처리하기 때문에 정확도가 평균 9.47% 향상되었다.



이 결과는 AIoT 자원 연계가 효율적으로 처리되기 때문이다. 처리 시간은 DNN 서비스 중 IoT 리소스를 평균 3.01% 더 빠르게 처리한다. 이 결과는 AIoT 리소스 연계 처리가 특정 크기 단위로 처리되기 때문이다.

4.2.2 시나리오 별 성능 평가

IoT 리소스가 머신러닝 알고리즘(LR, SVM, KNN)별로 세 가지 서로 다른 시나리오에 따라 표 5처럼 평가하였다.

- Scenario 1: 이 시나리오는 표 5와 같이 하드웨어가 AIoT 리소스의 모든 계산을 단일 블록체인으로 완전히 활용할 때의 평가 결과이다. 표 5에서 SVM은 일반적으로 대기 시간이 긴 단일 AIoT 리소스 처리에 필요한 광범위한 계산(벡터/매트릭스) 때문에 LR 및 KNN보다 정확했다. SVM은 IoT 리소스 처리를 빠르게 처리할 수 있지만, KNN은 많은 IoT 리소스를 처리할 때 IoT 리소스 지연 시간 측면에서 우수하다. 이 결과는 모든 AIoT 리소스를 완전히 활용해야 하므로 단일 블록이 더 높아야 하므로

AIoT 리소스를 처리하는 것이 효율성이 떨어졌다.

- Scenario 2: 이 시나리오는 AIoT 리소스의 멀티블록 처리를 평가한 결과이다. 여러 블록에서 AIoT 리소스를 처리할 때 AIoT 리소스 처리 결과는 시간 슬롯당 모든 기준선보다 훨씬 더 우수한 성능을 보였다. 그러나 AIoT 리소스의 처리 시간이 짧을수록 LR보다 SVM 및 KNN에서 AIoT 리소스의 처리 효율이 높아졌다.

- Scenario 3: 이 시나리오는 AIoT 리소스만 인식할 수 있는 추가 정보의 추출 및 처리 성능을 평가한다. AIoT 자원봉사자에서 AIoT 기타 리소스를 분류 및 처리하도록 구성되어 중복 추론으로 효율성을 향상시킨다. 이 시나리오를 통해 KNN과 SVM은 평균적으로 LR보다 더 나은 결과를 얻었다. AIoT 자원봉사자에서 IoT 추가 리소스를 분류할 때 SVM은 시간당 리소스 처리가 우수했다. LR은 높은 IoT 리소스 처리량과 대기 시간으로 인해 KNN 및 SVM보다 더 비효율이다. 이러한 결과는 여러 상관관계가 있는 AIoT 리소스 블록을 복잡한 멀티블록체인에 적용할 때 SVM이 잘 작동한다는 것을 보여준다.

표 5. 시나리오의 DNN 분석  
Table 5. Analyze variance DNN by scinario

Scenario	Round	Accuracy(%)			Precision(%)			Recall(%)			F-Score(%)		
		LR	KNN	SVM	LR	KNN	SVM	LR	KNN	SVM	LR	KNN	SVM
Scenario 1	1	82	89	94	87	90	96	86	89	94	85	89	93
	2	85	92	95	86	91	95	90	87	96	81	90	95
	3	81	86	96	85	89	93	88	90	91	84	86	92
	4	86	88	92	88	92	91	84	89	93	86	88	94
Scenario 2	1	82	89	94	87	90	96	86	89	94	85	89	93
	2	85	92	95	86	91	95	90	87	96	81	90	95
	3	81	86	96	85	89	93	88	90	91	84	86	92
	4	86	88	92	88	92	91	84	89	93	86	88	94
Scenario 3	1	82	89	94	87	90	96	86	89	94	85	89	93
	2	85	92	95	86	91	95	90	87	96	81	90	95
	3	81	86	96	85	89	93	88	90	91	84	86	92
	4	86	88	92	88	92	91	84	89	93	86	88	94

## V. 결론 및 향후 과제

최근에는 대기업을 중심으로 인공지능과 IoT 기술이 활용되면서 IoT 리소스를 효율적으로 처리하기 위한 다양한 연구가 진행되고 있다. 그러나 이기종 IoT 장치에서 수집된 AIoT 리소스를 관리하는 방법이 추가적으로 요구되고 있다. 본 논문에서는 AIoT 리소스 관리를 최적화하기 위한 가상화 자원 할당 기반의 AIoT 리소스 관리 기법을 제안하였다. AIoT 자원 관리의 모델 성능(정확성 및 견고성 측면에서)과 자원 비용을 동시에 최적화하기 위해 제안 방법은 라푸노프 최적화 이론을 사용하여 AIoT 리소스의 병목 현상, 처리 지연 및 네트워크 오버헤드를 최소화하였다. 성능평가 결과 AIoT 리소스 특성에 따라 주기적으로 그룹화하여 효율성을 실시간으로 처리하여 평균 7.17% 향상된 결과를 얻었다. AIoT 리소스만 인식할 수 있도록 AIoT 자원봉사자에서 추가 자원 정보만 추출하므로 정확도가 평균 9.47% 향상되었다. AIoT 리소스 연계 처리는 특정 크기 단위로 처리하기 때문에 처리 시간이 평균 3.01% 더 빠르게 처리되었다.

향후 연구에서는 본 연구에서 얻은 결과를 바탕으로 다양한 클라우드 환경에 적용하여 제안 방법의 성능을 평가할 예정이다.

## References

- [1] J. S. Lee, "A Study on the Effects of the Cooperative Philosophy between SMEs to the Cooperative Activities and Performance", *Journal of the Korea Convergence Society*, Vol. 8, No. 9, pp. 301-309, Sep. 2017.
- [2] C. Garrido-Hidalgo, D. Hortelano, L. Roda-Sanchez, T. Olivares, M. C. Ruiz, and V. Lopez, "Iot heterogeneous mesh network deployment for human-in-the-loop challenges towards a social and sustainable industry 4.0", *IEEE Access*, Vol. 6, pp. 28417-28437, May 2018. <https://doi.org/10.1109/ACCESS.2018.2836677>.
- [3] T. X. Tran and D. Pompili, "Joint task offloading and resource allocation for multi-server mobile-edge computing networks", *IEEE Transactions on Vehicular Technology*, Vol. 68, No. 1, pp. 856-868, Jan. 2019. <https://doi.org/10.1109/TVT.2018.2881191>.
- [4] L. Zhao, J. Wang, J. Liu, and N. Kato, "Optimal edge resource allocation in IoT-based smart cities", *IEEE Network*, Vol. 33, No. 2, pp. 30-35, Mar. 2019. <https://doi.org/10.1109/MNET.2019.1800221>.
- [5] F. Forooghifar, A. Aminifar, and D. Atienza, "Resource-aware distributed epilepsy monitoring using self-awareness from edge to cloud", *IEEE Transactions on Biomedical Circuits and Systems*, Vol. 13, No. 6, pp. 1338-1350, Dec. 2019. <https://doi.org/10.1109/TBCAS.2019.2951222>.
- [6] Y. Kang, et al., "Neurosurgeon: Collaborative intelligence between the cloud and mobile edge", *ACM SIGARCH Computer Architecture News*, Vol. 45, No. 1, pp. 615-629, Mar. 2017. <https://doi.org/10.1145/3093337.3037698>.
- [7] Z. Zhao, et al., "A Novel Framework of Three-Hierarchical Offloading Optimization for MEC in Industrial IoT Networks", *IEEE Transactions on Industrial Informatics*, Vol. 16, No. 8, pp. 5424-5434, Aug. 2020. <https://doi.org/10.1109/TII.2019.2949348>.
- [8] B. Yang, X. Cao, C. Yuen, and L. Qian, "Offloading Optimization in Edge Computing for Deep-Learning-Enabled Target Tracking by Internet of UAVs", *IEEE Internet of Things Journal*, Vol. 8, No. 12, pp. 9878-9893, Jun. 2021. <https://doi.org/10.1109/JIOT.2020.3016694>.
- [9] H. Gu, et al., "A Collaborative and Sustainable Edge-Cloud Architecture for Object Tracking with Convolutional Siamese Networks", *IEEE Transactions on Sustainable Computing*, Vol. 6, No. 1, pp. 144-154, Jan. 2021. <https://doi.org/10.1109/TSUSC.2019.2955317>.
- [10] D. Liu, et al., "FitCNN: A cloud-assisted and low-cost framework for updating CNNs on IoT

devices", Future Generation Computer Systems, Vol. 91, pp. 277-289, Feb. 2019. <https://doi.org/10.1016/j.future.2018.09.020>.

- [11] C. Gamanayake, L. Jayasinghe, B. K. K. Ng, and C. Yuen, "Cluster Pruning: An efficient filter pruning method for edge AI vision applications", IEEE Journal of Selected Topics in Signal Processing, Vol. 14, No. 4, pp. 802-816, May 2020. <https://doi.org/10.1109/JSTSP.2020.2971418>.
- [12] T. D. Nguyen, S. Marchal, M. Miettinen, H. Fereidooni, N. Asokan, and A.-R. Sadeghi, "D<sup>2</sup>IoT: A federated self-learning anomaly detection system for IoT", 2019 IEEE 39th International Conference on Distributed Computing Systems(ICDCS), Dallas, TX, USA, pp. 756-767, Jul. 2019. <https://doi.org/10.1109/ICDCS.2019.00080>.
- [13] S. Wang et al., "Adaptive Federated Learning in Resource Constrained Edge Computing Systems", IEEE Journal on Selected Areas in Communications, Vol. 37, No. 6, pp. 1205-1221, Jun. 2019. <https://doi.org/10.1109/JSAC.2019.2904348>.
- [14] J. Mills, J. Hu, and G. Min, "Communication-Efficient Federated Learning for Wireless Edge Intelligence in IoT", IEEE Internet of Things Journal, Vol. 7, No. 7, pp. 5986-5994, Jul. 2020. <https://doi.org/10.1109/JIOT.2019.2956615>.

김 옹 태(Yong-Tae Kim)



1984년 2월 : 한남대학교  
계산통계학과(공학학사)  
1988년 2월 : 숭실대학교  
전자계산학과(공학석사)  
2002년 2월 : 충북대학교  
전자계산학과(이학박사)  
2010년 10월 ~ 현재 : 한남대학교

멀티미디어학부 교수

관심분야 : 모바일 웹서비스, 정보 보호, 센서 웹, 모바일 통신보안

## 저자소개

정 윤 수 (Yoon-Su Jeong)



1998년 2월 : 청주대학교  
전자계산학과(공학사)  
2000년 2월 : 충북대학교  
전자계산학과(이학석사)  
2008년 2월 : 충북대학교  
전자계산학과(이학박사)  
2012년 2월 ~ 현재 : 목원대학교

게임소프트웨어공학과 교수

관심분야 : IoT/IIoT, 네트워크, 정보보안, 빅데이터, 암호학, 스마트팜