

협력 로봇의 동적 AI 작업 제어를 위한 테스트베드 구축 및 성능평가

장현진, 최지현, 김요한
동서대학교

20181602@g.dongseo.ac.kr, 20181630@office.dongseo.ac.kr, yhkim@dongseo.ac.kr

Testbed Implementation of Dynamic AI-Task Control for Collaborative Robots

Hyeonjin Jang, Jiheon Choi, Yohan Kim
Dongseo University

요약

MEC(Mobile Edge Computing) 기술 등의 발전으로 협력 로봇은 엣지에서 AI 작업을 신속하게 처리할 수 있게 되었다. 하지만 협력 로봇의 전력 제한, 배터리 용량 한계 등 가용 자원에 제약이 있어, 이를 지원하기 위해 AI 작업을 엣지와 원격 서버 중 적절한 위치에 배치해야 한다. 본 논문에서는 AI 작업을 동적으로 제어하기 위한 테스트베드를 구축하고 작업 수행시간에 대한 성능을 평가하였다.

I. 서론

4차 산업혁명의 도래와 함께 인공지능, 사물인터넷, 로봇 기술이 발전하면서 산업 현장에서 인간과 협력하여 AI 작업을 처리하는 협력 로봇 제어에 대한 연구가 많은 주목을 받고 있다[1]. 그러나 실제 산업 현장에서 사용되는 협동 로봇, AMR(Autonomous Mobile Robot), 드론과 같은 협력을 위한 로봇은 한정된 전력 및 배터리 용량 등으로 인한 가용 자원(i.e., GPU clock) 제한으로 인해 AI 작업을 단말에서 처리하는 데에 한계가 있다. 이러한 문제를 지원하기 위해 최근 MEC(Mobile Edge Computing) 기술을 바탕으로 엣지(edge) 또는 원격 서버에서 작업에 필요한 자원을 동적으로 제어하는 연구가 진행되었다[2]. 본 논문에서는 MEC 환경에서 협력 로봇의 AI 작업을 동적으로 제어하기 위한 테스트베드를 구축하고, AI 작업의 QoS(Quality of Service)를 보장하기 위해 가용 자원에 따라 작업의 위치를 결정하기 위한 동적 AI 작업 제어 시스템을 제안한다.

II. 협력 로봇을 위한 동적 AI 작업 제어 시스템

그림 1은 본 논문에서 고려하는 협력 로봇의 동적 AI 작업 제어 시스템을 보여준다. 작업 w 의 작업량을 a_w , 엣지와 원격 서버의 가용 자원을 각각 c_j , c_s 라고 할 때, 엣지에서 작업을 처리하는 경우 AI 작업의 수행시간 t_w^j 는 다음 식으로 나타낼 수 있다.

$$t_w^j = \frac{a_w}{c_j} \quad (1)$$

로봇의 가용 자원 한계로 인해 엣지에서 AI 작업의 처리가 어려울 경우에는 이를 원격 서버에서 지원해야 한다. 엣지에서 원격 서버로 AI 학습 데이터를 보내기 위한 전송 시간을 d_{j2s} , 원격 서버에서 엣지로 학습된 AI 모델을 보내기 위한 전송 시간을 d_{s2j} 라고 할 때, 원격 서버에서 작업을 처리할 경우 AI 작업의 수행시간 t_w^s 은 다음과 같이 계산된다.

$$t_w^s = \frac{a_w}{c_s} + d_{j2s} + d_{s2j} \quad (2)$$

사용자나 시스템에서 요구하는 AI 작업의 QoS 요구사항을 만족시키기 위해서는 엣지와 원격 서버 중 적절한 위치에 AI 작업을 배치해야 한다. 작업 w 의 QoS 요구사항을 q_w 라고 할 때 작업 배치를 위한 제약 조건은 다음과 같다.

$$t_w^j \leq q_w, t_w^s \leq q_w \quad (3)$$

q_w 는 사용자나 시스템에서 요구하는 작업 w 의 최대 허용 수행시간을 의미하며, 단말의 전력 제한 및 배터리 용량에 따라 결정된 가용 자원(i.e., GPU Clock)을 실시간으로 모니터링하여 QoS 요구사항을 만족시키도록 AI 작업의 동적 작업 제어 기법을 적용할 수 있다.

III. MEC 기반 협력로봇 제어를 위한 테스트베드 구축

원격 서버는 Tesla T4가 장착된 Workstation을 사용하였고, 엣지는 Jetson nano를 사용하여 협력 로봇의 동적 작업 제어를 위한 MEC 테스트베드를 구축하였다. GPU Clock에 따른 AI 작업의 수행시간을 측정하기 위해 Cocol28 데이터셋을 사용하여 YOLO nano 모델을 학습하도록 설정하였으며, Epochs=50으로 학습을 진행하였다. 표 1은 GPU Clock의 변화에 따른 Jetson Nano와 원격 Workstation에서의 작업 수행시간과 엣

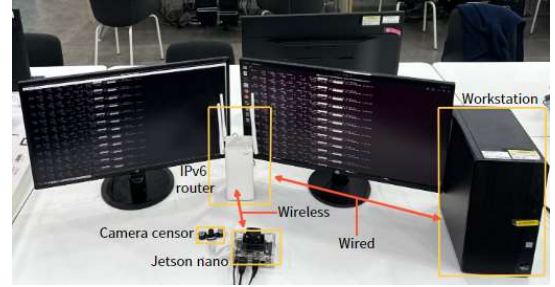


그림 1. 협력 로봇의 동적 작업 제어 시스템 구성

지-원격 간에 학습 데이터 및 모델을 전송하기 위한 전송시간을 나타낸다. 데이터의 전송은 유·무선 링크를 통해 SCP 기반으로 전송하였다.

표 1. GPU Clock에 따른 AI 작업 처리 시간 측정

		$t_w^j(s), t_w^s - d_{j2s} - d_{s2j}(s)$					$d_{j2s}(s)$	$d_{s2j}(s)$
GPU clock		min	25%	50%	75%	max		
Jetson Nano	YOLOv5	134.2	65.5	50	45.7	45	4.5	3.6
Workstation	YOLOv5	8.3	5.0	4.7	4.7	4.7		
	YOLOv8	7.9	5.8	5.8	5.8	5.4		

측정 결과는 전송 시간을 고려하더라도 원격 서버에서 AI 작업을 처리했을 때 소요되는 시간이 더 적은 것을 보여준다. 그러나, 실제 환경에서는 원격 서버에서 많은 수의 협력 로봇의 작업을 처리하기 때문에 원격 서버에서의 수행시간이 더 길어질 수 있다. 따라서 QoS를 고려하여 엣지와 원격 서버 간 적절한 위치에 AI 작업을 배치해야 한다.

III. 결론

본 논문에서는 MEC 환경에서 협력 로봇의 AI 작업 제어를 지원하기 위한 테스트베드를 구축하고 실험을 통해 작업 수행시간을 분석하였다. 또한, AI 작업의 QoS 요구사항을 만족할 수 있도록 가용 자원을 모니터링하여 작업을 어디서 처리할 것인지 결정하는 동적 작업 제어 시스템을 제안하였다. 향후 강화학습 등 인공지능 기술을 적용하여 자율적으로 AI 작업을 제어하는 알고리즘 연구를 진행할 계획이다.

ACKNOWLEDGMENT

“본 연구는 2024년 과학기술정보통신부 및 정보통신기획평가원의 SW중심대학사업의 연구결과로 수행되었음”(2019-0-01817)

참고 문헌

- [1] S. Kook, S. Baek, H. Nam and S. -L. Kim, “ML-based Vision Classification in the Smart Brewing Factory Testbed,” in Proc. IEEE International Conference on Communications Workshops (ICC Workshops), 2022.
- [2] Y. Kim and H. Lim, “Multi-Agent Reinforcement Learning-Based Resource Management for End-to-End Network Slicing,” in IEEE Access, vol. 9, pp. 56178-56190, 2021