

3/4축 협동 로봇을 위한 모바일 티칭 펜던트의 설계 및 구현

조현태[°], 구제훈^{*}, 신재정^{*}, 김범준^{**}

Design and Implementation of a Mobile Teaching Pendant for 3/4 Axes Collaborative Robots

Hyuntae Cho[°], Jehun Koo^{*}, Jaejung Shin^{*}, Beomjun Kim^{**}

요약

협동 로봇은 사람과 같은 공간에서 작업하면서 상호 협력하는 로봇으로 많은 산업 분야에서 활용되고 있다. 현재 활용되고 있는 협동 로봇의 대부분은 다관절 로봇을 기반으로 구축되어 있어, 복잡한 작업이 필요하지 않은 분야에서도 과도한 사양 및 비용들이 투입되고 있다. 본 논문에서는 단순한 작업을 수행하는데 활용될 수 있도록 직교 좌표 로봇과 SCARA 로봇과 같은 3/4축 로봇들과의 협동을 위한 모바일 티칭 펜던트를 설계하고 구현한다. 협동 로봇에 필요한 구성요소는 3/4축 로봇, 통신 및 로봇 제어를 위한 컨트롤 박스, 그리고 티칭 펜던트로 구성된다. 티칭 펜던트는 안드로이드 기반의 태블릿 PC에서 동작하도록 설계 및 구현하였다. 본 논문에서 설계 및 구현되는 티칭 펜던트의 역할은 1) 로봇의 동작을 제어하는 프로그램 코드를 작성할 수 있는 작업 환경, 2) 작성된 프로그램이 실행되면서 로봇을 제어하고 상태를 모니터링 할 수 있는 실행 화면, 그리고 3) 로봇을 제어하는데 필요한 환경을 설정하는 환경 설정으로 구성된다. 구현된 모바일 티칭 펜던트 앱은 무선 Wi-Fi를 통하여 원격의 컨트롤 박스와 데이터를 송수신하면서 제어 명령을 보내고 로봇의 상태를 실시간 모니터링 할 수 있도록 하였다. 또한 사용자 편의를 돕기 위해 다양한 부가 기능을 포함할 수 있도록 한다. 마지막으로 논문에서는 직교 좌표 로봇과 협동을 수행 및 검증하기 위해 충돌 테스트와 치킨 조리 로봇에 대한 프로그램을 모바일 티칭 펜던트로 작성하고 로봇과 실시간 통신하여 제어하는 테스트를 수행하였다.

키워드 : 협동 로봇, 직교 좌표, 스카라, 대쉬보드, 모바일 티칭 펜던트, 산업로봇

Key Words : Collaborative robots, Cartesial coordinate, SCARA, sashboard, mobile teaching pendant, industrial robots

ABSTRACT

Collaborative robots are robots that work in the same space as people and cooperate with each other, and are used in many industrial fields. Most of the collaborative robots currently in use are built based on articulated robots, resulting in excessive specifications and costs even in fields that do not require complex tasks. In this paper, we design and implement a mobile teaching pendant for cooperation with 3/4 axes robots such as a Cartesian coordinate and SCARA robot, enabling it to perform simple tasks. The components

※ 본 연구는 중소벤처기업부의 2023년도 창업성장기술개발사업 (팁스프로그램, (No. RS-2023-00258570))의 지원에 의한 연구임.

° First and Corresponding Author : Dept. of Digital Content, Tongmyong University, marine@tu.ac.kr, 정회원

* Dept. of Digital Media Engineering, Tongmyong University

** Dept. of Smart Software, Yonam Institute of Technology, bjkim@yc.ac.kr, 정회원

논문번호 : 202405-106-D-RN, Received May 17, 2012; Revised June 26, 2024; Accepted July 8, 2024

required for 3/4 axes collaborative robots consist of a robot, a control box for communication and control robots, and a teaching pendant. The paper designed and implemented the teaching pendant application to run on an Android-based tablet PC. The role of the teaching pendant is primarily to provide 1) a work environment for building control programs, 2) an execution display screen that can control the robot and monitor its status as the written program is executed, and 3) settings to configure the necessary parameters for robot control. The implemented teaching pendant app transmits and receives data to and from a remote control box via Wi-Fi, sending control commands and monitoring the robot's status in real time. Additionally, various additional functions can be included to enhance user convenience. Finally, in the paper, we tested a collision and performed and verified cooperation with a Cartesian coordinate robot by creating a program code for a chicken-cooking robot on a mobile teaching pendant.

I. 서 론

현대 산업의 빠른 발전과 함께 산업용 로봇의 수요 또한 지속해서 증가하고 있다. 전 세계 제조업에 투입된 산업용 로봇이 400만 대에 육박하고, 이와 동시에 한국은 노동자 1만 명당 로봇 대수로는 압도적인 1위이다^[1]. 또한, 그동안 단순 기계적 반복 기능을 하던 기존의 산업용 로봇과 달리 현대 산업용 로봇은 5G, 인공지능 등과 융합하여 더 복잡한 제조, 조립, 검사, 포장과 같은 작업, 인간이 할 수 없는 일을 수행하는 등 산업의 중요한 축을 담당하고 있다^[2,3]. 특히, 협동 로봇은 사람과 같은 공간에서 상호 협력하여 작업을 수행하는 로봇으로 많은 산업 분야에서 활용될 수 있다^[4,5]. 협동 로봇 글로벌 시장 보고서에 따르면 전 세계 협동 로봇 시장은 2020년에 9억 8,100 만 달러에서 연평균 성장률 41.8%로 증가하여 2026년에는 79억 7,200만 달러에 이를 것으로 전망하고 있다^[6,7].

글로벌 로봇 제작사인 Universal robots (덴마크), ABB (스위스), KUKA (독일), 레인보우 로보틱스 (한국), 두산 로보틱스 (한국), Rethink Robotics (미국) 등의 기업들이 협동 로봇을 제작 판매하고 있다. 이들이 판매하는 협동 로봇은 대부분 다관절 로봇을 이용하여 협동 작업을 수행하도록 설계되어 있다. 다관절 로봇은 모든 축이 회전축으로 구성되어 있으며 공간상에서 6축 자유도 (degree of freedom)를 가지고 있어 섬세하고 다양한 응용 분야에 활용될 수 있다.

하지만, 다관절 로봇은 3/4축 로봇에 비하여 상대적으로 고가이고, 실제 산업 현장이나 응용 분야에서는 6축 자유도를 요구하지 않고 3/4축의 자유도만 요구하면서 간단한 작업을 수행하는 경우가 있다^[8,9]. 또한 산업용으로 빈번히 활용되는 직교 좌표 로봇이나 SCARA (selective compliance assembly robot arm or selective compliance articulated robot arm) 로봇을 그대로 이용

하여 산업 현장에서 협동 로봇으로 대체하고자 하는 수요도 함께 존재한다.

본 논문에서는 상대적으로 단순하거나 반복적인 작업을 수행하는데 활용되는 직교 좌표 로봇과 SCARA 로봇과 같은 3/4축 협동 로봇 위한 모바일 티칭 펜던트 (teaching pendant)를 설계하고 구현한다. 많은 협동 로봇 시스템의 경우에 로봇과 협동을 수행하기 위해서 대시보드 형태의 티칭 펜던트를 사용하고 있다. 이 대시보드 티칭 펜던트를 사용하여 로봇을 제어하기 위한 프로그램을 작성하고 로봇의 상태를 실시간으로 확인하면서 협동 작업을 수행한다.

본 논문에서는 별도의 대시보드 하드웨어를 사용하지 않고, 안드로이드 태블릿 PC에 애플리케이션을 구현하여 모바일 티칭 펜던트를 구현한다. 설계된 모바일 티칭 펜던트는 기존의 대시보드 티칭 펜던트가 가지는 위치적 제약과 불편한 사용자 인터페이스를 가진 고정형 티칭 패널의 단점을 극복하도록 하였다. 모바일 티칭 펜던트는 1) 로봇의 동작을 제어하는 프로그램 코드를 작성할 수 있는 작업 환경, 2) 작성된 프로그램이 실행되면서 로봇을 제어하고 상태를 모니터링 할 수 있는 실행 화면, 그리고 3) 로봇을 제어하는데 필요한 환경을 설정하는 환경 설정으로 구성된다. 작업환경을 통해 사용자는 프로그램에 대한 지식이 없더라도 그래픽 사용자 인터페이스를 통해 로봇이 동작하는 제어 프로그램을 간편하고 직관적으로 작성할 수 있다. 이러한 작업 환경은 로봇의 행동을 쉽고 유연하게 변경할 수 있는 협동 로봇 시스템으로 만들어 준다. 협동 로봇은 사람과 함께 같은 공간에서 작업하기 때문에 사람과의 충돌을 빠르고 정확하게 안내해 주어야 한다. 제한된 티칭 펜던트에서는 로봇과 사람이 충돌할 경우에 실시간으로 충돌 상황을 알려주고, 로봇을 실시간으로 제어할 수 있게 한다. 또한, 로봇을 제어 및 프로그램하기 위한 사용자 인터페이스 및 통신 프로토콜을 포함하도록 설계되었

다. 그리고 본 논문에서는 티칭 펜던트와 로봇이 상호 협동할 수 있도록 티칭 펜던트를 통하여 치킨 조리 프로그램을 작성하고 실시간으로 로봇을 제어하여 모바일 티칭 펜던트의 기능들을 검증하였다.

본 논문의 구성은 다음과 같다. 2장에서는 협동 로봇과 관련한 연구들에 대하여 조사를 수행하고, 3장에서는 모바일 티칭 펜던트를 설계하고 구현한다. 4장에서는 실험을 통하여 티칭 펜던트와 로봇의 상호작용을 검증하고 5장에서는 향후 과제와 함께 결론을 맺는다.

II. 관련 연구

협동 로봇은 인간과 협력하여 작업을 수행하는 로봇으로, 산업, 의료, 서비스 등 다양한 분야에서 연구되고 있다. 협동 로봇에 관한 연구는 안전성, 효율성, 인간-로봇 상호작용 (human-robot interaction) 등을 중심으로 진행되고 있다.

- 안전성: 협동 로봇은 인간과 가까운 거리에서 작업하기 때문에 안전성이 매우 중요하며, 로봇과 사람 사이에 사고를 방지하기 위해 로봇의 센서 기술과 인공지능을 활용한 인식 기술이 활용되고 있다. 협동 로봇은 카메라, LiDAR, 초음파 센서 등을 사용하여 주변 환경을 인식하고, 잠재적 충돌을 피하기 위한 회피 계획 및 제어 기술들을 사용한다^{10,11}. 또한, 영상과 인공지능을 활용해 인간의 동작을 미리 예측하고, 이에 맞춰 로봇의 행동을 제어하는 연구가 있다. 이러한 연구를 통해 인간과 로봇 상호간에 부드러운 상호작용 또는 협력을 수행할 수 있다^{12,13}.
- 인간-로봇 상호작용: 작업자와 로봇 간의 의사소통 및 협력을 통하여 생산성을 높이기 위해 자연어 처리 및 제스처 인식에 대한 연구들이 진행되고 있다. 자연어 처리 및 음성 인식은 인간의 음성 명령을 이해하고 이에 따라 로봇이 움직이고 행동하게 하는 기술이다. 예를 들어, 특정 작업 지시를 음성으로 주고받을 수 있다¹⁴. 제스처와 같은 비언어적 신호를 인식하여 로봇이 인간의 의도를 파악하고 적절히 대응하는 연구가 진행되고 있으며 최근 YOLO와 같은 기술도 사용되고 있다¹⁵.
- 협력 작업 계획: 협동 로봇이 인간과 효율적으로 작업을 나누어 작업하도록 계획 및 프로그램하는 것이다. 티칭 펜던트를 이용하여 로봇이 작업할 내용을 프로그램하는 것이 이에 해당한다. 로봇과 인간이 각자 맡은 작업을 효율적으로 수행할 수 있도록 역할을 분담하고, 작업의 진행 상황을 동기화하

도록 한다¹⁶. 더 나아가, 작업 환경이나 조건이 변화할 때, 로봇이 실시간으로 계획을 조정하여 대응할 수 있는 기술에 대한 연구들도 선행되었다¹⁷. 또한 [18] 연구에서는 기존의 티칭 펜던트들은 사용이 용이하지 않다는 실험 결과를 도출하였고, 사용 흐름에 대한 패턴화, 직관적인 조그 (jog)와 프로그래밍 방식에 대한 개선 필요 등의 개선안을 제시하였다.

인간과 로봇이 상호 협력하면서 공동의 작업을 수행하기 위해서는 앞서 언급한 3가지 요구사항들을 만족할 수 있어야 한다. 그리고 사람과 로봇이 협력할 수 있도록 중간 매개체 역할을 하는 것이 티칭 펜던트이다. 로봇의 충돌이 감지되면 로봇은 즉시 동작을 멈추고, 현재의 상황을 티칭 펜던트를 통하여 사용자와 공유하여 문제를 해결하여야 한다. 또한 협력 작업 계획을 작성하는 것을 포함하여 로봇의 상태를 모니터링하고 로봇에 제어 장치를 실시간으로 관찰 및 제어하면서 자연스러운 상호 작용을 수행할 수 있어야 한다. 본 논문에서는 사용자에게 친화적인 안드로이드 기반의 티칭 환경을 제공하여 협력 작업 계획 작성하고 자연스러운 상호 작용을 가능하게 하며, 안전성을 제공하는 모바일 티칭 펜던트를 설계하고 구현하는데 중점을 둔다.

앞서 소개한 많은 협동 로봇 회사들은 자사의 로봇에 특화된 티칭 펜던트 또는 대시보드를 함께 출시하고 있다. 그림 1은 독일 KUKA사의 협동 로봇 티칭 펜던트의 사용 사례를 보여준다. 그림에서 보는 바와 같이 모바일 형태의 티칭 펜던트 모양을 유지하고 있으나, 로봇을 티칭하기 위한 독립적인 하드웨어와 소프트웨어가 함께 제공되고 있다. 티칭을 위한 소프트웨어가 하드웨어에 종속적이기 때문에 확장성이나 비용 측면에서 많은 단점들이 존재한다.



그림 1. 독일 KUKA사의 협동 로봇을 위한 보편적인 티칭 펜던트
 Fig. 1. Conventional teaching pendant for Germany KUKA's collaborative robots.

III. 모바일 티칭 펜던트의 설계 및 구현

티칭 펜던트란 로봇을 제어하는 데 사용되는 휴대용 장치로 로봇의 위치 지정하고 각 축을 제어하며 로봇 프로그램을 실행하거나 긴급 정지하는 등의 기능을 수행한다¹⁹⁾. 본 논문에서는 조그, 제어 프로그램 작성 등을 포함한 모든 사용자 인터페이스를 사용자 친화적으로 만들고 초보자도 쉽고 간편하게 로봇을 티칭 할 수 있는 모바일 티칭 펜던트를 설계한다.

3.1 사용자 인터페이스 구조

로봇과 협동을 위한 모바일 티칭 펜던트 앱은 안드로이드 태블릿 PC에서 동작할 수 있는 앱으로 구현이 되고, 그림 2와 같이 3개의 주요 액티비티 (activity)로 구성된다. 앱이 실행되면 메인 액티비티에서 3가지 작업 중 하나의 작업을 선택할수록 설계한다.

3개의 화면 간에 이동을 중개하는 메인 액티비티에서는 통신을 위한 연결 설정, 로봇을 티칭하기 위한 프로그램을 작성하는 작업 환경(make activity), 만들어진 로봇의 프로그램을 실행하는 실행 화면 (play activity), 로봇 실행에 필요한 파라미터 들을 변경할 수 있는 환경 설정 화면 (setup activity)으로 이동할 수 있는 기능을 포함한다. 또한 각 화면마다 원하는 화면으로 넘어갈

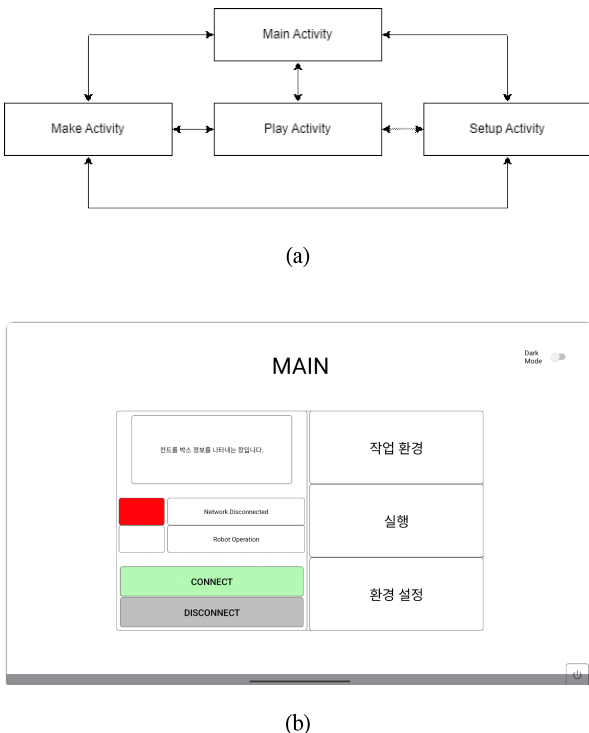


그림 2. 안드로이드 티칭 펜던트 액티비티의 구조: (a) 액티비티 구조 및 (b) 메인 액티비티 사용자 인터페이스
Fig. 2. Teaching pendant activities on Android: (a) activity structure and (b) user interface for main activity.

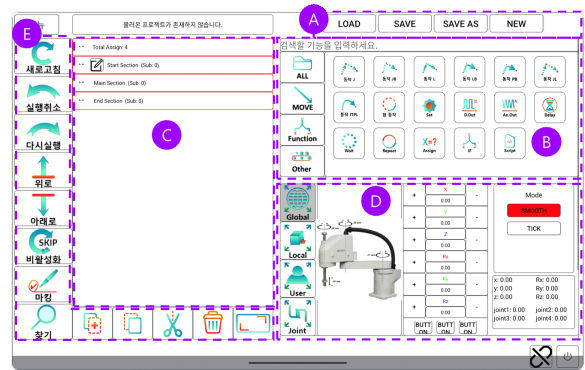


그림 3. 작업 화면 사용자 인터페이스
Fig. 3. User interface for a make activity.

수 있는 버튼은 별도로 구성된다. 또한, 사용자는 항상 자신이 원하는 화면을 어느 액티비티에서든지 선택 및 이동할 수 있다.

3.2 작업 액티비티 (make activity)

작업 환경은 로봇 티칭 작업을 진행하고 로봇 프로그램을 만드는 화면으로, 로봇의 티칭에 필요한 기능들이 배치되어 있다. 즉, 사용자가 이 작업 액티비티를 통하여 로봇을 제어하기 위한 프로그램 코드를 그래픽 사용자 인터페이스를 통해 작성할 수 있다. 그림 3은 작업 액티비티를 위한 사용자 인터페이스를 보여주며 각 영역 별로 기능은 다음과 같다. 영역 A는 협동 로봇을 제어하고 동작하기 위한 코드를 작성하기 위해 새로운 프로젝트를 만들고, 저장하고, 기 저장된 프로그램 코드를 불러는 기능들을 포함하고 있다. 본 논문에서 설계 및 구현한 모바일 티칭 펜던트에서는 협동 로봇을 제어하기 위한 명령어들은 JSON 파일로 저장되어 관리된다.

영역 B는 협동 로봇을 제어하고 필요한 정보를 로봇으로부터 획득하기 위한 명령어들 모아 놓은 부분이다. 영역 B의 명령어들은 각각 독립적인 하위 명령어 및 정보들을 필요로 한다. 예를 들어, Move 명령어는 로봇의 각 축이 이동할 좌표값 또는 거리, 이동 속도 등을 포함하며, 로봇의 그리퍼 (gripper)를 동작시키기 위한 디지털 출력 (D.Out)의 경우는 디지털 포트 번호 및 상태 값을 포함한다. 이러한 하위 정보를 포함하게 하기 위해, 본 논문에서는 제어 명령어 삽입 과정에서 명령어 객체 (command objects)를 사용하는데 그 구조는 그림 4와 같다. 명령어 객체는 최상위 명령어 클래스를 상속 받아 구현하게 되고, 어떤 명령 클래스의 객체인지와 필드의 종류를 통해 명령어를 구분하게 된다. 명령어 객체는 로봇 티칭에 필요한 명령어를 표현한 객체이며, 해당 기능에 필요한 데이터들을 저장할 수 있는 필드와 하위에 지식 명령어를 가질 수 있는 리스트를 가지고

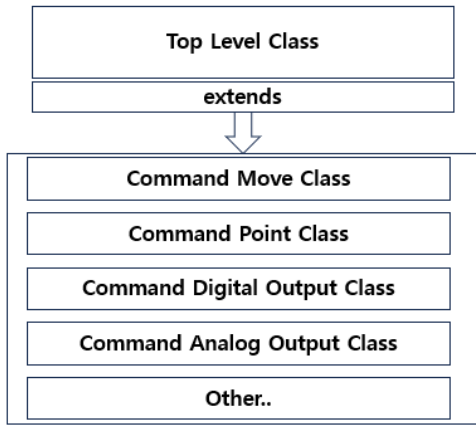


그림 4. 명령어 클래스 구조 (move 명령어)
Fig. 4. Structure of a command Class (move command).

있다. 이 명령어 클래스는 명령어 카테고리별로 다른 상속의 형태를 가진다.

그림 5는 협동 로봇 제어 프로그램 코드에 명령을 삽입하는 과정을 보여준다. 명령을 추가하기 위해 사용하는 영역 B에서 필요한 명령 아이콘을 선택하면, 새로운 명령어 트리가 생성된다. 이 트리의 최상위 계층은 아이콘에 해당하는 명령이 된다. 가장 많이 사용되는 명령이 Move 명령이고, Move 아이콘을 클릭할 경우 Move에 대한 명령어 트리가 생기고, 이 트리의 하위 리프(leaf)에 필요한 정보들이 추가로 연결된다. 그런 다음 세팅 명령을 통해 Move 명령어가 필요한 정보를 추가로 기입할 수 있다. 이 세팅 명령어에는 명령어의 별명, 좌표계의 종류, 축의 움직임 속도, 가속도, x, y, z 축의 좌표값 또는 pitch, yaw, roll 또는 관절의 각도 값을 기입할 수 있다. 이때, 절대 좌표값이 아닌 로봇의 현재의 실시간 좌표값을 기반으로 이동하여야 할 경우 로봇으로부터 좌표값을 얻어 올 수 있다. 이 정보들은 그림 4의 하위 포인트 (point) 객체에 선택되어 리스트로 연결된다. 필요한 정보가 모두 채워지면 이 프로그램 스크립트는 협동 로봇 제어 프로그램 코드에 추가된다. 여러 개의 타겟 지점으로 연속적으로 이동하여야 할 경우에는 반복 작업을 통해 프로그램 코드에 추가할 수 있다.

로봇을 제어하는데 필요한 영역 B 명령은 표 1에서 보는 바와 같다. 표에서 Linear 계열은 주로 3축 로봇, Joint 계열의 명령은 주로 4축 SCARA 로봇에 사용된다. 물론 Linear 계열의 3축 정보도 SCARA에 사용될 수 있다.

표 1의 명령어들이 각각 필요한 모든 정보를 포함하게 되면, 그림 6과 같이 다양한 명령어들이 모여 로봇을 제어하는 티칭 프로그램이 형성고, 그림 3의 영역 C에

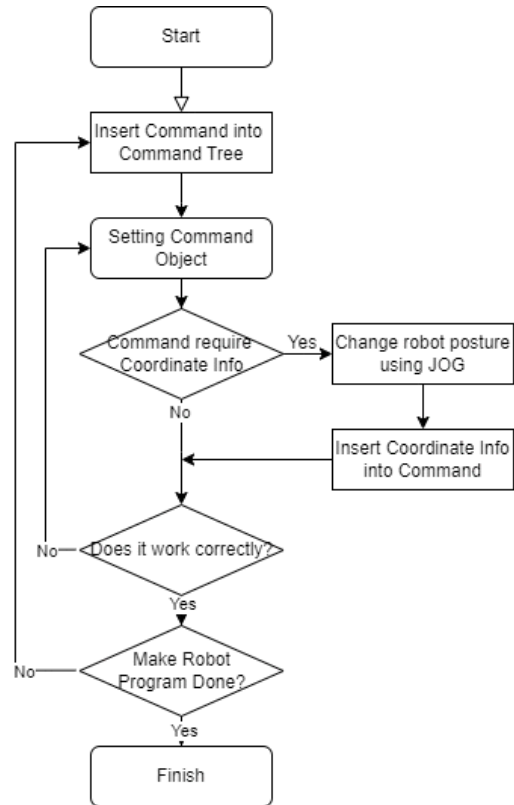


그림 5. 명령 입력 과정
Fig. 5. Procedure for command input.

표 1. 프로그램 가능한 제어 기능
Table 1. Control commands to be able to program.

Commands	Description
MoveJ	Move Joint
MoveJB	Move Joint Blend
MoveL	Move Linear
MoveLB	Move Linear Blend
MovePB	Move Point Blend
MoveJL	Move Joint with Linear Input
MoveITPL	Move Interpolation
Circle	used for circle behavior with three points
Set	set parameters used in robots
D.Out	On/Off digital pins
A.Out	set analog value to pins
Delay	delay (unit milliseconds)
Wait	wait until the D.IN (digital input) or A.IN (analog input) is reached to some status
Repeat	loop
Assign	define variables
IF	if condition statement
Script	make user defined scripts

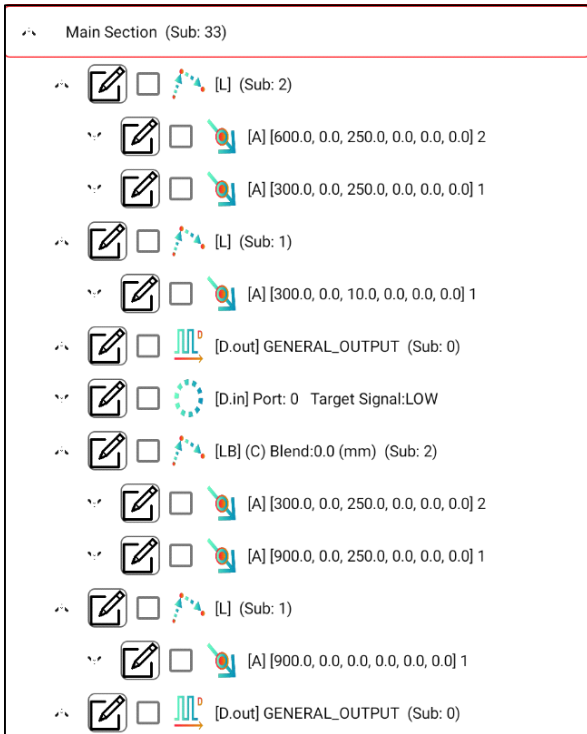


그림 6. 프로그램 코드 트리 예시
Fig. 6. Example of a program code tree.

나타나게 된다. 작성된 프로그램 코드에서 각각의 명령어는 리스트에 순차적으로 삽입되어, 실행 액티비티를 통해 실행되면서 제어 메시지를 로봇으로 전달한다.

좌표값이 요구되지 않는 Delay, Wait, D.Out 등의 명령어들은 각각의 상황에 맞는 윈도우를 통하여 필요한 정보를 입력하고 수행하여 필요시 로봇으로 정보를 전하고 제어한다.

작업 화면의 또 다른 중요한 기능은 조그 기능은 로봇을 X, Y, Z, yaw, roll, pitch 축으로 각각 수동으로 조작할 수 있는 기능이다. 조그 기능은 그림 3의 영역 C를 통해 사용할 수 있다. 조작 가능한 단위는 각 축으로 1mm이며, 4가지 좌표계를 사용하여 로봇의 축을 이동시키는 것이 가능하다. 기본적으로 리니어 모드로 이동하는 것이 가능하며, SCARA 로봇을 사용할 경우에는 각도를 통하여 이동하는 것도 가능하다.

3.3 실행 액티비티 (Play Activity)

그림 7은 실행 액티비티를 통하여 로봇을 제어하기 위해 수행되는 실행 순서도이다. 실행 화면(play activity)에서는 만들어진 로봇 프로그램의 명령어들을 순차적으로 실행하는데 이때 명령어가 통신이 필요하면 직교 좌표 로봇의 컨트롤 박스에 전송 후 응답을 받으면 다음 명령어를 실행하고, 통신이 필요하지 않으면 기능 수행 후 다음 명령어를 실행한다. 그 과정에서 실행 에

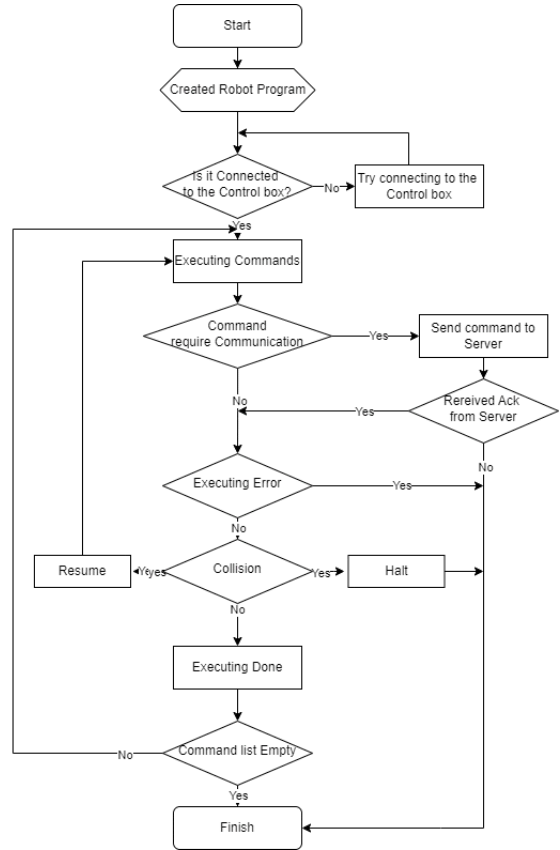


그림 7. 실행 액티비티 동작 과정
Fig. 7. Operation procedure of a play activity.

러가 발생 시 실행을 중지하고, 실행 도중 충돌 발생 시 사용자에게 프로그램 흐름 재개(resume) 혹은 프로그램 중지(halt)를 선택할 수 있게 한다.

그림 8은 설계된 실행 액티비티에 대한 GUI 화면으로써, 로봇으로 데이터를 보내면서 제어하는 실행 화면이다. 실행 중인 명령어는 계층에 따라 연두색, 노란색으로 표시하여 사용자에게 시각적으로 강조하며, 컨트롤 박스에서 온 데이터들을 수신하고 로그로 출력한다.

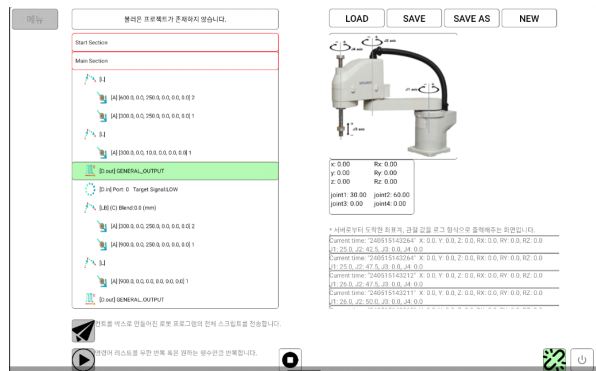


그림 8. 명령어를 실행중인 실행 화면
Fig. 8. Play activity executing commands.

3.4 환경 설정 액티비티 (Setup Activity)

로봇의 제어에는 로봇의 민감도, 사용자 지정 좌표계에 대한 정보 등 다양한 데이터들이 필요하고 이를 설정해 주어야 한다. 애플리케이션 내부적으로도 보안을 위한 비밀번호, 통신을 위한 IP 및 포트 설정 등 다양한 설정이 필요하다. 그림 9의 환경 설정 화면에서는 이러한 다양한 파라미터 설정을 진행한다. 여기서 민감도 등 로봇 제어에 필요한 값들은 컨트롤 박스와 연결됨과 동시에 컨트롤 박스에 전송된다. 또한 사용자가 설정한 파라미터 값들은 YAML로 저장되어 애플리케이션 실행 시 자동으로 이를 불러와 적용시킨다. 그 외에 비밀번호, 충돌감지 감지, 시리얼통신, 입출력 등에 대한 다양한 환경 설정을 할 수 있도록 설계되었다.

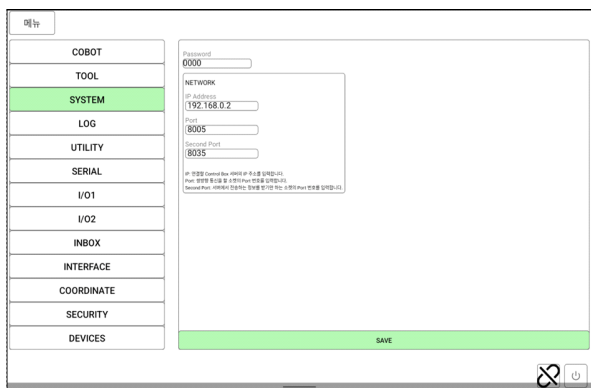


그림 9. 환경 설정 화면
Fig. 9. User interface for a setup activity.

3.5 통신 프로토콜

그림 10은 사용자, 모바일 티칭 펜던트, 컨트롤 박스, 그리고 3/4축 협동 로봇 사이의 연결 및 통신 구조를 보여준다. 모바일 티칭 펜던트와 로봇의 컨트롤 박스가 통신을 하기 위해 통신 채널을 표 2와 같이 두 개로 분리하여 운영한다. 첫 번째 채널 (TCP1)은 포트 번호 8000번을 사용하여 일반적인 송신 및 수신 기능을 수행하고, 두 번째 채널 (TCP2)은 포트번호 8030번을 사용하여 로봇의 현재 좌표값, 로봇 연결 여부 및 충돌 상태를 0.5초마다 수신하는데 사용된다. 통신을 위해 사용되는 IP는 버전4가 활용되며, 로컬 네트워크 또는 외부 네트워크 연결이 모두 가능하다. 컨트롤 박스의 경우 서버로 동작을 하며 Xenomi와 EtherCat을 사용하여 실

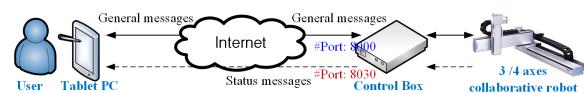


그림 10. 통신을 위한 네트워크 구조
Fig. 10. Network architecture for communication.

표 2. 채널별 통신 기능
Table 2. Function by channel.

Channel	Function
CH1 (TCP1)	Jog(Global, Local, User, Joint)
	Move
	Set
	Digital Out
	Analog Out
	Wait
CH2 (TCP2)	Robot operation status
	Robot co-ordinate

시간 동작을 할 수 있도록 설정되었다.

모바일 티칭 펜던트와 컨트롤 박스는 상호 메시지를 주고 받기 위해 소켓 모듈을 사용하여 TCP 프로토콜 통신을 사용할 수 있도록 하였다. 앞서 언급한대로 컨트롤 박스는 서버로 동작하고 있으며, 로봇이 동작하는 동안에는 서버로서의 역할을 수행할 수 있고 티칭 펜던트는 클라이언트로 접속하여 데이터를 송수신한다. 티칭 펜던트 앱은 서버와 통신을 수행하기 위하여 소켓의 입출력 스트림을 사용하고, 통신에 필요한 프로토콜을 구현한다.

그림 11은 채널 1을 통한 일반 메시지를 송수신할 때 사용하는 메시지의 형식을 보여준다. 해당 메시지는 TCP 페이로드에 포함되는 부분으로 main command 1바이트, sub command 1바이트, 데이터 페이로드 N 바이트로 구성된다. 그리고 마지막 1바이트는 에러 체크를 위한 용도로 예약하였다. main command로는 모두 45개의 메시지 명령어가 있으며, Move, Joint, Debug, 및 각 명령어의 ACK (acknowledgement) 메시지들도 포함한다. 이 main command의 MSB (most significant bit) 1비트는 통신 방향을 나타내기 위한 비트로 사용된다.

Sub command는 각 main command의 하위 정보를 포함하기 위한 프로토콜이다. 예를 들면, 그림 12와 같

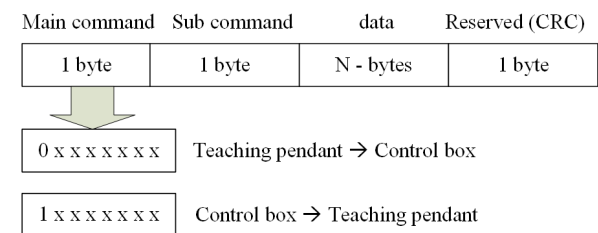


그림 11. 일반 메시지 형식
Fig. 11. general purpose message format.

이 main command의 타입이 Move이면, sub command는 J(joint), JB (joint blend), L (linear), LB (linear blend), PB (point blend), JL (joint with linear input), IPTL (interpolation) 등의 sub command를 가질 수 있다. 그리고 Data 페이로드에는 type과 데이터 페이로드가 추가될 수 있으며, type은 좌표계의 형식을 설정할 수 있다. Data 페이로드에는 실제 좌표값, 속도, 등의 값들이 포함된다.

Move 명령을 받은 로봇의 컨트롤 박스는 main command를 0x04 (main command에 대한 ACK)를 포함하여 sub command와 데이터를 포함하여 회신한다. 데이터 페이로드에는 로봇 동작 중 (0x00), 동작 완료 (0x01), 오동작 (0x02), 로봇 충돌(0x03)에 대한 정보를 포함하여 회신할 수 있다.

채널 2를 통하여 로봇의 상태를 주기적으로 전송하는 패킷의 형태는 그림 13과 같다. 데이터 포맷은 일반 메시지 형식과 같으나 주기적으로 컨트롤 박스가 로봇의 좌표와 충돌 상태를 데이터 페이로드에 포함하여 전달한다.

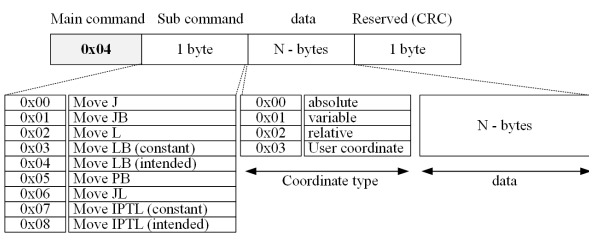


그림 12. Move 명령어를 위한 메시지 형식
Fig. 12. Message format for a move command.

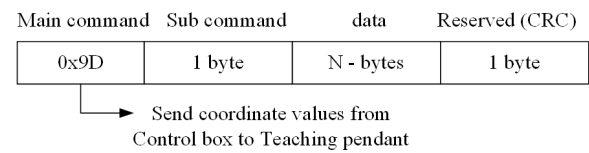


그림 13. 상태 메시지 형식
Fig. 13. Status message format.

IV. 실험 및 검증

4장에서는 본 논문에서 설계 및 구현한 모바일 티칭 펜던트가 사용자가 의도한 대로 데이터를 로봇의 컨트롤 박스로 전송하고 로봇이 동작하는지 확인 및 검증한다.

4.1 실험 환경

티칭 펜던트 앱의 검증을 위하여 본 논문에서는 직교 좌표 로봇을 이용하여 치킨을 튀기는 요리를 하는 응용 프로그램을 구현하였다. 치킨을 조리하는 프로그램의 흐름은 그림 14와 같다. 조리 준비를 위한 바구니는 2개가 있고 튀김 냄비가 2개가 있다. 로봇 팔이 움직이는 좌표 값은 총 9개가 있으며, 이들을 움직이며 조리를 하며 총 4단계를 거친다. (a) 초벌 튀김 및 (b) 반복 튀김의 튀김 과정을 거치고, (c) 조리된 치킨은 저장되는 바구니에 담긴 후, (d) 주문이 오면 다시 재 튀김을 수행한다. 로봇 팔이 움직이는 과정에서 튀김 기름망을 통해서 옮기는데 이를 집고 내려 놓고 하는 작업은 티칭 펜던트의 디지털 신호 출력 (D.OUT)을 통해 제어한다.

그림 15는 태블릿 PC의 티칭 펜던트 앱에서 구현한 치킨 조리 프로그램 코드의 일부분을 보여준다. 그림의 영역 B에서 보는 바와 같이 명령어 코드들의 그래픽 형태로 나열되어 있는 것을 볼 수 있다. 이 코드에 대한 실제 동작은 그림 16에서 흐름도를 통해 확인할 수 있다.

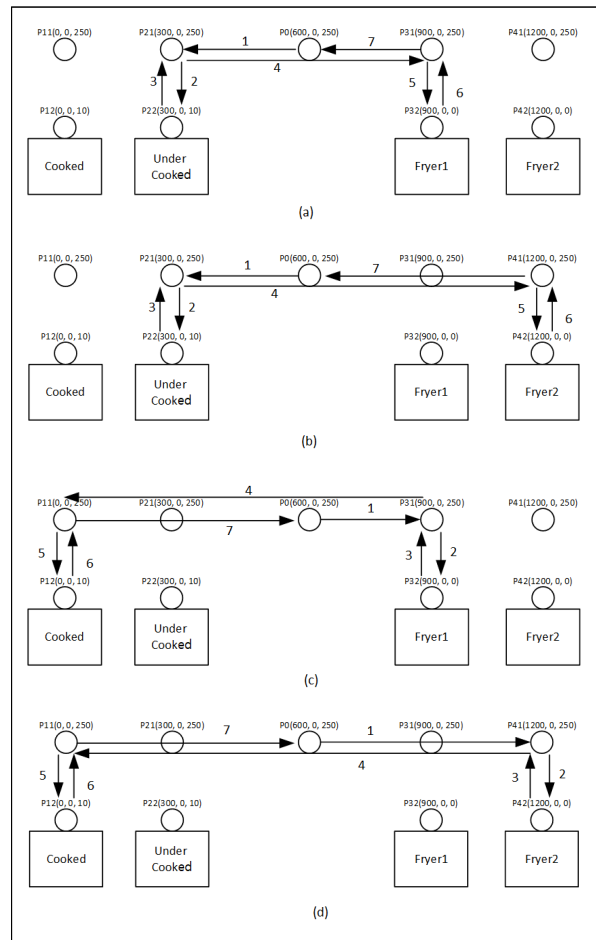


그림 14. 치킨 조리 과정 (로봇 암의 이동 순서)
Fig. 14. Chicken cooking process (movement sequence of robot arm).

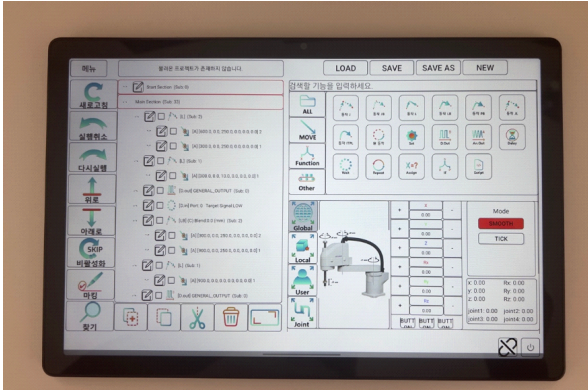


그림 15. 치킨 요리 응용 프로그램
Fig. 15. Chicken cooking program.

그림 14에서 설명한 바와 같이 4개의 조리 단계가 있고, 로봇과 협동하는 사용자는 4가지 조리중에 1개의 과정을 선택하여 로봇의 작업을 지시한다. 그림 16 (a)는 초벌 튀김에 대한 과정, (b)는 반복 튀김, (c)는 조리된 치킨을 바구니에 담는 과정 (d)는 재 튀김하는 과정에 대한 순서도이다. 순서도에서 Gripper를 동작시키는 작업은 D.Out (digital out) 명령을 조합하여 구현한다. 로봇의 컨트롤 박스와 통신하기 전에 로컬 서버 예물

레이터를 구현하여 본 논문에서 설계한 티칭 팬던트 앱이 데이터를 제대로 송신하는지 확인한다. 로컬 서버는 티칭 팬던트 앱을 구현하는 과정에 프로토콜 테스트를 위하여 함께 윈도우상에서 구축되었다. 해당 서버는 애플리케이션에서 전송한 데이터를 분석하여 어떤 Type의 명령어와 데이터가 도착했는지 출력하여 보여준다. 그림 17는 티칭 팬던트 앱에서 로컬 서버에 접속한 후 치킨 요리 프로그램을 동작시키는 과정 중 수신되는 패킷을 와이어샤크를 통하여 캡처한 것이다. (a)는 Move Linear 명령어를 보낼 때, 필요한 패킷 구조이다. main command, sub command type, data, 및 reserved 바이트로 구성되어 있다. 데이터의 내용은 float 데이터 형으로 전달되며, 축의 이동 속도, 가속도, x,y,z 좌표값, yaw, roll, pitch, 그리고 3개의 관절에 대한 값을 포함하고 있다. (b)는 Move Linear 명령을 로봇이 수행한 후에 ACK를 회신하는 패킷을 캡처한 것이다. 세 번째 바이트의 0x01은 동작이 정상적으로 완료되었음을 의미한다. 본 논문에서 설계한 패킷 페이로드 이외의 부분은 TCP 프로토콜의 헤더 정보들이다.

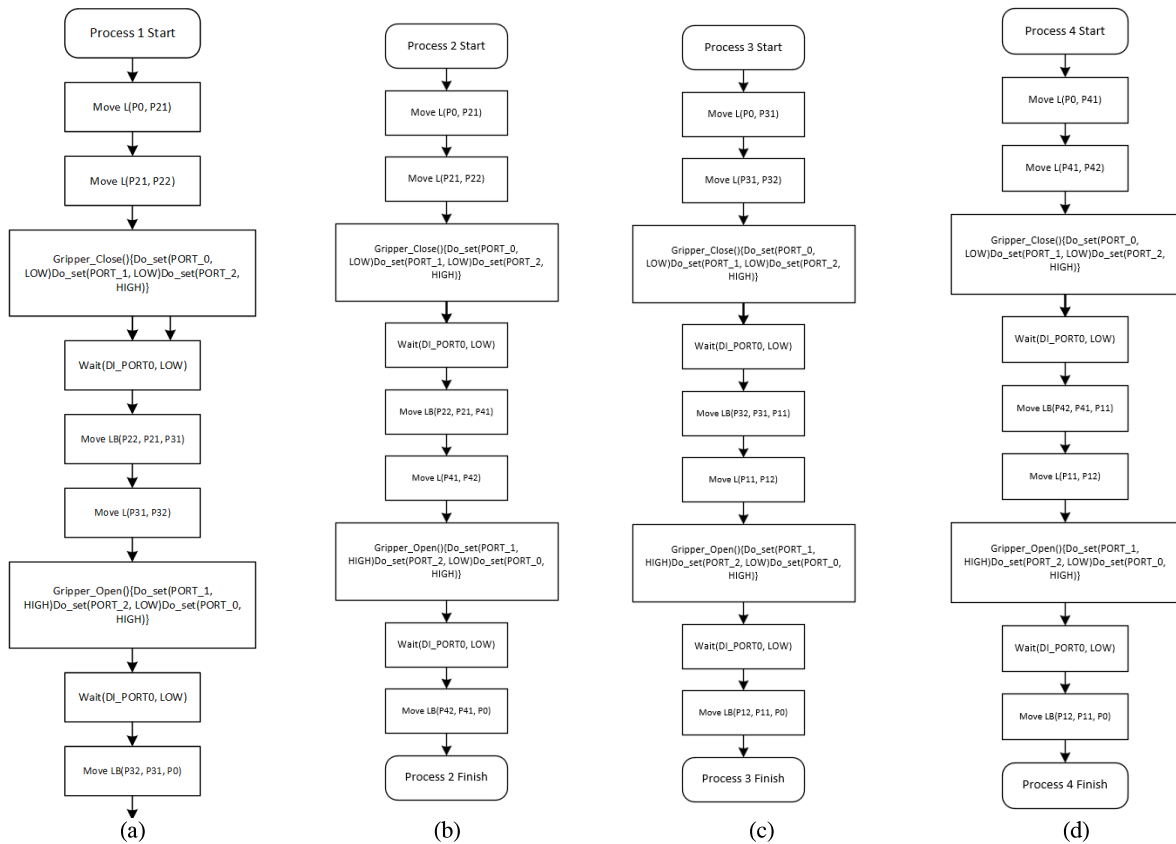


그림 16. 치킨 조리 과정에 대한 순서도
Fig. 16. Flowchart for chicken cooking process.

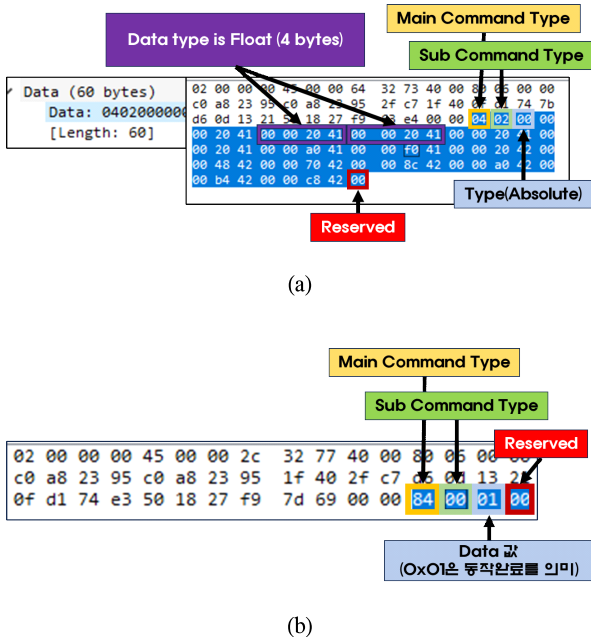


그림 17. 데이터 송수신 검증: (a) Move Linear 패킷 및 (b) ACK 패킷
 Fig. 17. Verification of data transmission and reception: (a) Move Linear packet and (b) ACK packet.

4.2 직교 로봇 제어 검증

구현된 티칭 펜던트 앱을 통하여 실제 로봇에 장착된 컨트롤 박스와 연결하여 로봇을 티칭 및 제어한다. 검증에 사용된 로봇은 국내 제작 직교좌표 로봇으로 로봇 제어 프로그램 및 통신 프로그램은 컨트롤 박스에 구현되어 있다. 컨트롤 박스는 ASUS Computer 사의 빌트인 인텔 셀러론 컴퓨터로 그림 18과 같으며, Ubuntu 운영체제로 동작하며 EtherCat과 Xenomi를 통하여 실시간 통신 및 동작을 보장하도록 설계되었다.

그림 19는 티칭 펜던트를 통하여 치킨 조리 프로그램

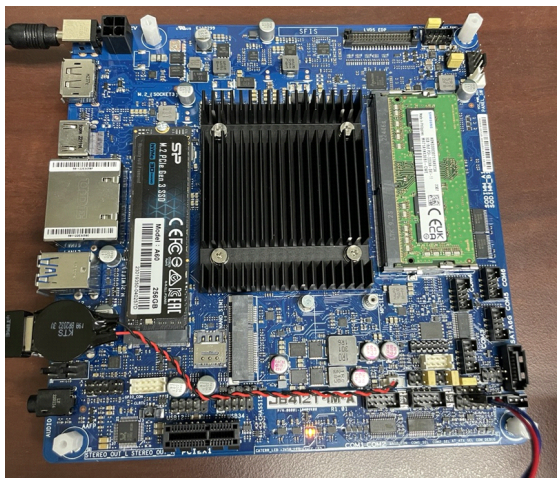


그림 18. 컨트롤 박스 (ASUS J6412T-IM-A)
 Fig. 18. Control box (ASUS J6412T-IM-A).

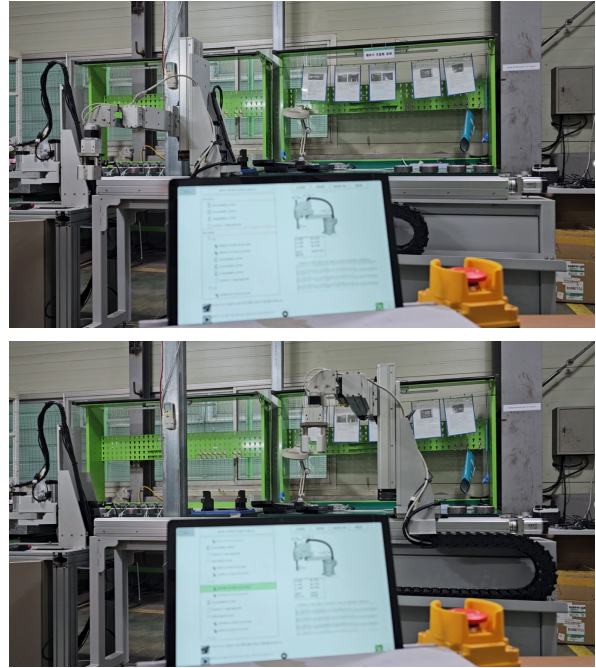


그림 19. 명령어를 수신받고 동작하는 로봇
 Fig. 19. A robot that receives commands and operates.

램을 실행하면, 로봇이 의도한 대로 움직이는 것을 보여준다. 본 논문의 실험에서는 치킨 조리 로봇의 기능에 대하여 분석하고 이를 티칭 펜던트 앱을 통하여 직교좌표 로봇을 구동 및 제어하였다. 본 논문에서 설계 및 구현한 모바일 티칭 펜던트는 태블릿 PC 상에서 구현되어, 쉽게 다양한 응용분야에 활용될 수 있다. 실제 3축 직교 좌표 로봇과 4축 SCARA 로봇이 사용되는 산업 분야에서 협동 로봇으로 프로그래밍하여 사람과 함께 작업을 수행할 수 있을 뿐만 아니라, 단순한 작업을 수행하는 고가의 다관절 협동 로봇을 대체할 수 있다. 다양한 응용에 활용되기 위해서는 다른 협동 로봇과 마찬가지로 본 논문에서 설계 구현된 모바일 티칭

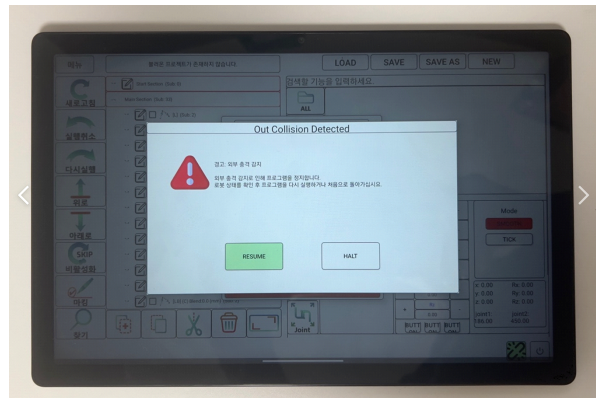


그림 20. 충돌 상태에 대한 검증
 Fig. 20. Verification of collision status and UI.

트를 통하여 3/4축 로봇을 제어하는 프로그램을 구현하여야 한다. 이로 인하여 쉽게 3/4축 로봇을 자유롭게 프로그래밍할 수 있는 장점이 있다.

4.3 충돌 감지 및 알람

협동 로봇은 같은 공간에서 사람과 함께 동작하기 때문에 사람과 로봇이 충돌할 경우 안전상의 문제가 발생할 수 있다. 이러한 문제를 방지하기 위해 로봇이 충돌을 감지할 경우 실시간으로 충돌 상황을 알려주고 로봇을 실시간으로 제어할 수 있어야 한다. 본 논문에서 사용되는 직교 좌표 로봇은 충돌을 검출하기 위해 티칭 팬던트가 송신한 이동 거리와 속도를 참고하여 로봇이 실제 이동하는 거리와 속도를 계산하여 그 차이를 기반으로 충돌을 감지한다. 충돌이 감지되면 로봇은 즉시 멈추고 감지된 충돌을 0.5초 마다 보내는 상태 알림 메시지에 충돌 패킷을 실어 전송한다. 이를 수신한 티칭 팬던트 앱은 그림 20과 같이 티칭 팬던트에게 전송한다. 사용자는 상황을 판단하고, 로봇을 다시 동작 시킬 것인지 완전 중지할 것인지를 판단할 수 있게 한다.

V. 결 론

본 논문에서는 저가형 직교 좌표 로봇과 SCARA 로봇을 위한 모바일 티칭 팬던트 앱을 개발하여, 기존의 고정적이고 불편한 패널 형식의 티칭 도구의 단점을 보완하여 더 위치적인 부분에서 자유롭게 사용자 친화적으로 로봇과 협동할 수 있는 툴을 제공하였다. 실험을 통해 안드로이드 기반 애플리케이션은 무선 통신을 활용하여 무리 없이 직교 좌표 로봇의 컨트롤 박스와 데이터를 송, 수신하여 로봇을 동작시킬 수 있음을 검증하였다. 산업 로봇의 수요가 계속해서 증가하는 현대 사회에서 이러한 간편하고 강력한 티칭 팬던트의 장점은 사용자들로 하여금 더 쉽고 편리한 로봇 티칭 환경을 제공하고, 효과적인 산업 로봇의 운영에 도움을 줄 것으로 예상된다.

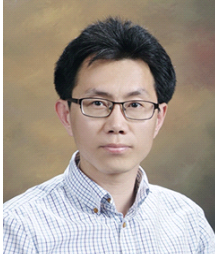
향후 과제로 직접 로봇과 통신을 수행하지 않더라도 프로그램을 구현한 후에 해당 프로그램이 정상적으로 동작하는지 자체적으로 검증 및 시뮬레이션 하는 기능을 추가, 로봇의 움직임 좌표 및 현재 각축의 위치를 티칭 팬던트에도 동기화하고, 협동 직교 로봇의 우수성을 검증하기 위해 다양한 성능 지표에 대하여 평가를 실시 할 것이다.

References

- [1] International Federation of Robotics, “*World robotics 2023*,” IFR World Robotics 2023 Report, 2023.
(<https://ifr.org/ifr-press-releases/news/world-robotics-2023-report-asia-ahead-of-europe-and-the-americas>)
- [2] J. Lee and J. Woo, “Trends and future of 5G-based advanced manufacturing robot technology,” *Inf. and Commun. Mag.*, vol. 38, no. 1, pp. 14-20, 2021.
(<https://www.kics.or.kr/data/Magazine/2021-01/EBook/ICM03.pdf>)
- [3] B. Kim and H. Kim, “A design of a smart control system for industrial automation equipment,” *J. Korea Academia-Industrial Cooper. Soc.*, vol. 18, no. 4, pp. 677-684, 2017.
(<https://doi.org/10.5762/KAIS.2017.18.4.677>)
- [4] F. Vicentini, “Collaborative robotics: A survey,” *J. Mechanical Design*, vol. 143, no. 4, p. 040802, 2021.
(<https://doi.org/10.1115/1.4046238>)
- [5] D. Kragic, J. Gustafson, H. Karaoguz, P. Jensfelt, and R. Krug, “Interactive, collaborative robots: Challenges and opportunities,” *IJCAI*, pp. 18-25, 2018.
(<https://doi.org/10.24963/ijcai.2018/3>)
- [6] Korea Innovation Foundation, “*Collaborative robot market*,” Global market trend report, Aug. 2021.
(<https://www.innopolis.or.kr/fileDownload?titleId=178533&fileId=1&fileDownType=C¶mMenuId=MENU00999>)
- [7] MarketsandMarkets, “*Collaborative Robot (Cobot) Market*,” 2020.
(https://www.marketsandmarkets.com/Market-Reports/collaborative-robot-market-194541294.html?gad_source=1&gclid=CjwKCAjwnK60BhA9EiwAmpHZw63IcHhErAdXO_R7xHS3xONCMEulYMHvaZ0oucVF3v1Kn-bM-V0qCxoC0c4QAvD_BwE)
- [8] A. Hentout, M. Aouache, A. Maoudj, and I. Akli, “Human-robot interaction in industrial

- collaborative robotics: A literature review of the decade 2008-2017,” *Advanced Robotics*, vol. 33, no. 15-16, pp. 764-799, 2019.
(<https://doi.org/10.1080/01691864.2019.1636714>)
- [9] M. Al-Quraishi, E. Natarajan, and S. S. A. Ali, “The expanding role of artificial intelligence in collaborative robots for industrial applications: A systematic review of recent works,” *Machines*, vol. 11, no. 1, pp. 1-25, 2023.
(<https://doi.org/10.3390/machines11010111>)
- [10] X. Wang, Z. Keményb, J. Vánca, and L. Wang, “Human-robot collaborative assembly in cyber-physical production: Classification framework and implementation,” *CIRP Annals*, vol. 66, no. 1, pp. 5-8, 2017.
(<https://doi.org/10.1016/j.cirp.2017.04.101>)
- [11] G. Haddeler, A. Aybakan, M. C. Akay, and H. Temeltas, “Evaluation of 3D LiDAR sensor setup for heterogeneous robot team,” *J. Intell. & Robotic Syst.*, vol. 100, no. 2, pp. 689-709, 2020.
(<https://doi.org/10.1007/s10846-020-01207-y>)
- [12] L. Rozo, S. Calinon, D. G. Caldwell, P. Jimenez, and C. Torras, “Learning physical collaborative robot behaviors from human demonstrations,” *IEEE Trans. Robotics*, vol. 32, no. 3, pp. 513-527, 2016.
(<https://doi.org/10.1109/TRO.2016.2540623>)
- [13] O. C. Görür, B. Rosman, F. Sivrikaya, and S. Albayrak, “Social cobots: Anticipatory decision-making for collaborative robots incorporating unexpected human behaviors,” in *Proc. 2018 ACM/IEEE Int. Conf Human-Robot Interaction*, pp. 398-406, 2018.
(<https://doi.org/10.1145/3171221.3171256>)
- [14] E. Matheson, R. Minto, E. G. G. Zampieri, M. Faccio, and G. Rosati, “Human-robot collaboration in manufacturing applications: A review,” *Robotics 2019*, vol. 8, no. 4, p. 100, 2019.
(<https://doi.org/10.3390/robotics8040100>)
- [15] P. Jiang, D. Ergu, F. Liu, Y. Cai, and B. Ma, “A review of yolo algorithm developments,” *Procedia comput. sci.*, vol. 199, pp. 1066-1073, 2022.
(<https://doi.org/10.1016/j.procs.2022.01.135>)
- [16] G. Dumonteil, G. Manfredi, M. Devy, A. Confetti, and D. Sidobre, “Reactive planning on a collaborative robot for industrial applications,” in *Proc. 12th Int. Conf. Inf. in Control, Autom. and Robotics*, vol. 2, pp. 450-457, 2015.
(<https://doi.org/10.5220/0005575804500457>)
- [17] W. Wang, R. Li, Z. M. Diekel, and Y. Jia, “Robot action planning by online optimization in human-robot collaborative tasks,” *Int. J. Intell. Robotics and App.*, vol. pp. 161-179, 2018.
(<https://doi.org/10.1007/s41315-018-0054-x>)
- [18] J. Dong, W. Kwon, D. Kang, and S. W. Nam, “A study on the usability improvement process of teaching devices for manipulator of collaborative robot,” in *Proc. Symp. KICS*, pp. 1078-1079, 2021.
(https://www.dbpia.co.kr/pdf/pdfView.do?nodeId=NODE10587378&googleIPSandBox=false&mark=0&minRead=5&ipRange=false&b2cLoginYN=false&aiChatView=B&readTime=5-10&icstCls=010000&isPDFSizeAllowed=true&accessgl=Y&language=ko_KR&hasTopBanner=true)
- [19] M. Oh, S. Lee, T. Kim, K. Lee, and J. Kim, “Design of a teaching pendant program for a mobile shipbuilding welding robot using a PDA,” *Comput.-Aided Design*, vol. 42, no. 3, pp. 173-182, 2010.
(<https://doi.org/10.1016/j.cad.2009.09.005>)

조 현 태 (Hyuntae Cho)



2003년 2월 : 한국해양대학교
자동화정보공학부 졸업
2005년 2월 : 부산대학교 컴퓨
터공학과 석사
2011년 2월 : 부산대학교 컴퓨
터공학과 박사

2011년~2012년 : 부산대학교 연구교수
2012년~2021년 : KAIST (재)스마트IT융합시스템연
구단 연구교수
2021년~현재 : 동명대학교 조교수
<관심분야> 임베디드 시스템, IoT, 스마트센서, 경
량 인공지능, 체외진단기기, 스테레오 비전
[ORCID:0000-0002-8951-0239]

구 제 훈 (Jehum Koo)



2019년 2월~현재 : 동명대학교
디지털미디어공학부 융합미
디어전공 학사과정
<관심분야> 협동로봇, 인공지
능, 사물인터넷, 모바일 프
로그래밍

신 재 정 (Jaejung Shin)



2019년 2월~현재 : 동명대학교
디지털미디어공학부 융합미
디어전공 학사과정
<관심분야> 컴퓨터네트워크,
사물인터넷, 모바일 프로그
래밍, UI/UX

김 범 준 (Beomjun Kim)



2012년 2월 : 부산대학교 정보
컴퓨터공학부 공학사
2014년 2월 : 부산대학교 전자
전기컴퓨터공학과 공학석사
2021년 2월 : 부산대학교 정보
융합공학과 공학박사

2021년~현재 : 연암공과대학교 조교수
<관심분야> 클라우드컴퓨팅, 데이터베이스, 딥러닝,
임베디드 시스템, 로봇, 스마트팩토리
[ORCID:0000-0002-2945-1054]