

LW-RCP 와 Odroid 기반의 Machine Vision 을 이용한 Ball and Plate 시스템 구현에 관한 연구

A Study on the Implementation of a Ball and Plate System using LW-RCP and Machine Vision based on Odroid

박주현¹, 방희승¹, 이영삼^{1,*}
(Juhyun Park¹, Heeseung Bang¹, and Young Sam Lee^{1,*})

¹Department of Electrical Engineering, Inha University

Abstract: In this paper, we propose the architecture of a ball-and-plate system as an educational platform that can help students comprehensively study control engineering, robotics, machine vision, and embedded systems. In addition, we describe how to implement the sub-systems and present an example of an education plan using the proposed system. In the proposed system, we use an LW-RCP (Light Weight – Rapid Control Prototyping) system, which is lab-built, to help students implement the control algorithm. An RCP system makes it easy for students who are not familiar with u-Controller or the C language to implement controllers. We also use Odroid, a low-cost SBC (single-board computer), and a single camera to calculate the position of the ball using a machine vision algorithm. In this structure, the measurement subsystem is more durable compared with contact-based sensing systems and students can test their own machine vision algorithms for better measurements. We adopt the Stewart platform, a type of parallel robot, to generate roll-and-pitch actuation for the ball-and-plate system. By adopting the Stewart platform, we can handle more complicated control problems for movement that requires more than two degrees of freedom. Additionally, we implement an LQ controller, which is a typical model-based controller, and apply it to the ball-and-plate system. Finally, through experiments, we illustrate the usefulness of the proposed system.

Keywords: education platform, ball-and-plate system, machine vision, stewart platform, RCP, u-controller, SBC, odroid, LQ

I. 서론

최근 대학교의 공학 교육과정에서는 학생들에게 효율적 학습경험을 제공하기 위해 설계과목을 통한 이론과 실습이 병행되는 학습 방법들을 적용한다[1]. 이러한 흐름에 따라 자동제어, 임베디드 시스템 설계 등의 제어와 로봇공학 과목에서도 제어이론을 배우고 동시에 실습 도구를 사용해 이론을 검증하는 경험을 통해 학습의 효율을 극대화한다[2-4].

고전적으로 제어 학습을 위한 실습 도구로 도립진자나 ball-and-plate 등의 플랫폼들이 사용된다. 그러나 제어 학습과 연구에 기초적으로 사용되는 직선형 도립진자는 레일에 고정되어 움직임에 제약이 있다. 이로 인해 학습 과정 중 제어기 설계가 잘못된 경우 레일 길이의 한계에 의한 파손 위험이 있다[5]. 레일 고정의 제한이 없는 도립진자 시스템으로 슬립링을 사용한 회전형 도립진자 시스템[6], 전방향 주행체 기반 2-DOF 도립진자 시스템[7]들에 대한 연구들이 있지만, 여전히 제어를 위해 넓은 기동 범위가 필요하기 때문에 한 실험실에서 많은 학생이 함께 사용하기 어렵다. 반면 ball-and-plate 시스템은 상대적으로 작은 기동 공간 내에서 실험이 가능할 수 있으며 제어기 설계를 잘못하여 발산하여도 사람과 제어기에 심한 손상을 입히지 않기 때문에 학습을 위한 실습 도구로 적합하다[8].

Ball-and-plate 시스템은 고전적인 ball-and-beam 문제로부터 확장된 전형적인 다변수 비선형 시스템으로 평평한 판 위에 공을 올려놓고 그 판을 기울여줄 수 있는 구동기를 조작해 공의 움직이는 시스템이다[9]. Ball-and-plate 시스템을 이용한 연구는 터치패널 같은 접촉식 센서를 사용하는 시스템과 카메라와 같은 비접촉식 센서를 사용한 시스템으로 나뉜다 [10,11]. 접촉식 센서는 제어주기가 짧아 빠르게 공의 위치를 얻을 수 있고 제어 안전성이 높지만, 판의 흔들림에 의한 위치 측정 노이즈가 있고 공과 센서의 잦은 충돌로 센서가 손상되는 문제가 있다. 반면 비접촉식 센서인 카메라는 공이 판과 접촉하지 않아도 공의 위치를 얻을 수 있고 공에 의해 센서가 손상되는 문제가 없다. 하지만 카메라 기반의 실시간 머신비전 시스템을 구현하기 위해 보다 높은 사양의 시스템을 요구한다. 이와 같은 머신비전 시스템을 구현하기 위한 카메라는 높은 프레임에 가져야 하고, 이를 실시간으로 처리하기 위해 처리 지연 시간이 짧은 대용량 혹은 병렬 연산 장치를 사용한다. 안정적인 머신비전 시스템을 구현하는 연구들로 고가의 스마트 카메라와 컴퓨터를 사용하는 시스템[13], 고속 웹카메라와 FPGA를 이용한 시스템[14]들이 제안되었다. 그러나 많은 머신비전 기반의 ball-and-plate 시스템들은 전체 시스템 구성 비용이 많이 들고 제어기 설계를 위해 전문 보

* Corresponding Author

Manuscript received January 29, 2020; revised February 18, 2020; accepted February 26, 2020

박주현: 인하대학교 전기공학과 대학원생(populi414@gmail.com, ORCID[®] 0000-0001-8700-7755)

방희승: 인하대학교 전기공학과 대학원생(hs.bang0718@gmail.com, ORCID[®] 0000-0003-4331-1952)

이영삼: 인하대학교 전기공학과 교수(lys@inha.ac.kr, ORCID[®] 0000-0003-0665-1464)

※ 이 논문은 2017년도 정부(교육부)의 재원으로 한국연구재단의 지원을 받아 수행된 연구임(No. 2017R1D1A1B03029578).

드들의 사용 방법을 익혀야 하는 어려움 때문에 실습 도구로 사용하기 어렵다. 이런 문제들을 해결하는 방법으로 저가의 SBC를 영상 처리기로 사용하는 ball-and-plate 시스템이 제안되었다[15]. 또 전문보드 사용에 대한 사전지식 없이도 제어를 구현할 수 있게 하는 연구로 GUI 기반의 제어기 설계와 실시간 제어 실험 환경을 제공하는 RCP 시스템이 제안되었다.

RCP 시스템은 제어기 설계와 설계한 제어기의 실제 제어 성능 검증을 빠르게 작업할 수 있게 하는 개발환경이다. RCP 시스템의 핵심 역할은 코드 생성 기능으로 전문 보드의 사용이 익숙하지 않은 학생들에게 오류가 발생하기 쉬운 핸드 코딩 작업을 없애고 보다 쉬운 블록 코딩 등의 방법을 제공하여 제어기 설계, 검증의 작업에 집중할 수 있게 해주는 것이다. 이러한 특성 때문에 많은 제어 교육과 관련된 연구들도 RCP를 사용한 교육적 목적의 시스템을 제안한다[16,17].

본 논문에서는 학생들의 효율적 학습을 위하여 low-level 프로그래밍 능력이 없이도 제어 시스템 설계가 가능한 RCP 시스템과 저비용으로 학생들에게 친숙한 운영체제 기반의 개발환경을 제공하는 SBC가 사용된 ball-and-plate 시스템 구조를 제안한다. 기존 SBC를 이용한 머신비전 시스템 연구는 비교적 저가의 하드웨어 구성이 가능했으나 마이크로컨트롤러를 이용한 핸드 코딩 작업이 필요하고 느린 영상처리 작업 속도로 인해 제어 성능의 한계가 있었다. 그러나 본 논문에서 제안하는 시스템은 마이크로컨트롤러 사용법을 익히지 않아도 제어기 설계가 가능하며 더 뛰어난 하드웨어 성능을 바탕으로 더 나은 제어 성능을 기대할 수 있어 학생들이 실습 도구로 사용하기 적합하다. 또한 정확한 제어주기에 따른 공의 위치 전달을 위한 SBC와 마이크로컨트롤러의 구성은 더 안정적으로 공의 위치를 이용한 실험이 가능하다. 나아가 구동부로 6-DOF의 병렬로봇의 일종인 Stewart platform 구동부를 채택한다. 이로 인해 학생들에게 기존의 시스템들보다 복잡한 움직임을 갖는 로봇시스템을 학습할 수 있게 한다. 제안하는 시스템의 전체 구조는 제어기의 역할을 하는 RCP 시스템, 독립적 센서의 역할을 하는 SBC가 사용된 영상처리 시스템, 구동부의 역할로 다양한 움직임을 가능한 Stewart platform으로 이루어진다.

II. LW-RCP의 소개

제안하는 ball-and-plate 시스템에서 제어기 설계를 위해 사용하는 RCP는 연구실에서 자체 개발된 오픈소스 하드웨어 기반의 LW-RCP이다[17]. LW-RCP는 그림 1과 같이 Windows 운영체제가 설치된 PC의 Matlab/Simulink 프로그램과 오픈소스 하드웨어인 Arduino Due 보드로 구성된다. 두 하부 시스템은 실시간 제어 실험과 시뮬레이션에 필요한 데이터를 high-speed USB를 통해 주고받는다. LW-RCP의 다른 RCP와의 차이 중 하나는 코드 생성 없이 Simulink에서 LW-RCP를 지원하는 블록 라이브러리를 이용하여 실시간 제어 실험 환경을 구축함에 있다. LW-RCP의 동작 순서는 다음 그림 2와 같다.

먼저 Simulink와 Due 보드의 초기화 작업을 진행하고 Due 보드가 특정 센서부로부터 필요한 데이터 값을 받아오면 이 값을 high-speed USB 장치를 통하여 PC의 Simulink로 전송한다.

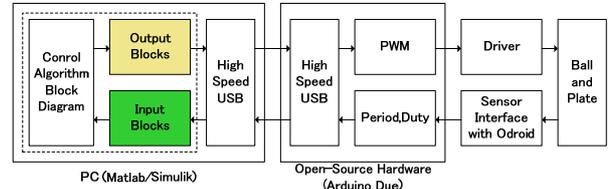


그림 1. LW-RCP와 제안하는 ball and plate 시스템의 연결도.
Fig. 1. Connection diagram between LW-RCP and proposed ball and plate system.

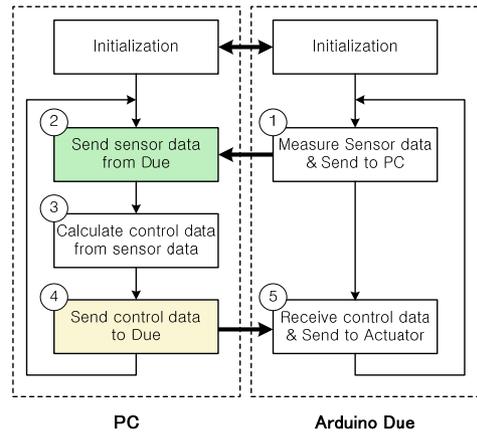


그림 2. LW-RCP 시스템의 동작 흐름도.
Fig. 2. Flowchart of the LW-RCP system.

Simulink는 받아온 센서 값을 센서 값과 Simulink 모델을 이용하여 제어 연산을 수행한다. 계산된 제어 연산 결과는 Due 장치에 송신되고 출력장치를 통해 출력한다.

LW-RCP의 Simulink에서 실시간 제어연산을 하기위해 센서 값을 입력 받아야 하고 실제 사용하는 시스템으로 출력 가능해야 한다. 이를 위해 LW-RCP에서 제공하는 기능에 따라 C 코드로 제작된 I/O블록 라이브러리를 사용한다. 그림 3의 좌측 1열 같이 입력을 위한 블록들과 2열의 출력을 위한 블록들로 구분된다. LW-RCP에서 입력으로는 PWM 파형의 주기와 duty를 읽을 수 있는 블록과 ADC 값을 읽어오는 블록, Digital input 등의 값을 확인할 수 있는 블록들이 있다. 출력

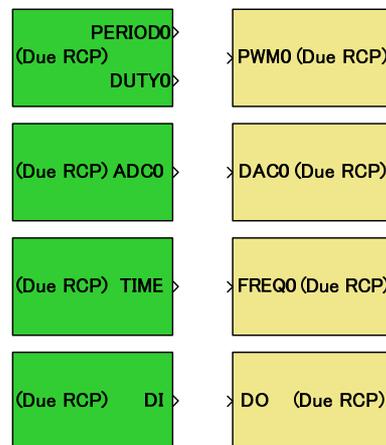


그림 3. Simulink에서 사용가능한 LW-RCP 블록들.
Fig. 3. LW-RCP blocks available on simulink.

으로는 PWM 파형을 만들어주는 블록과 DAC 블록, Digital output을 만들어주는 블록 등이 있다. 위 블록들과 제어를 위한 블록들을 사용하여 시스템에 따른 제어를 설계해 전체 시스템을 제어한다.

III. 제안하는 시스템 구조

제안하는 시스템의 하드웨어는 그림 4와 같이 배치한다. 학생들에게 제어공학, 로봇공학, 머신비전 등의 학습 환경을 제공하기 위해 본 시스템은 그림 5의 개념도 같이 세 가지의 하부 시스템을 사용한다. 센서부로는 머신비전 알고리즘을 실습할 수 있는 카메라, SBC, 마이크로컨트롤러 구성의 머신비전 시스템을 사용한다. 또 제어부로 제어기 설계와 실시간 제어 실험이 가능한 LW-RCP를 사용한다. 마지막 구동부는 학생들에게 로봇의 제어기 설계에 필요한 운동역학을 분석할 수 있게 하는 Stewart platform을 사용한다. 제안하는 시스템 구조는 학생들에게 필요한 영역의 학습을 가능하게 한다.

학습을 위한 실습도구로 제안되는 본 시스템은 사용자의 선택에 따라 시스템 모델을 설계할 수 있으며 시스템의 움직임에 따라 필요한 머신비전 알고리즘을 적용하고 실시간 제어 실험을 할 수 있다. 본 논문에서는 제안하는 시스템의 안정적 동작 사례로 다음과 같은 상황을 가정하여 머신비전 알고리즘과 시스템 모델을 설정하고 제어 실험을 실시하여 그 결과를 보인다. 머신비전 알고리즘은 병렬로봇의 다양한 움직임을 고려하여 판의 기울기 움직임과 수평이동까지 고려한 판 움직임에 강한 머신비전 알고리즘으로 설계한다. 또 제어를 위한 시스템 모델은 고전적인 ball-and-plate 모델을 사용하여 2-DOF의 움직임인 판의 평면에 대한 각 축의 기울기

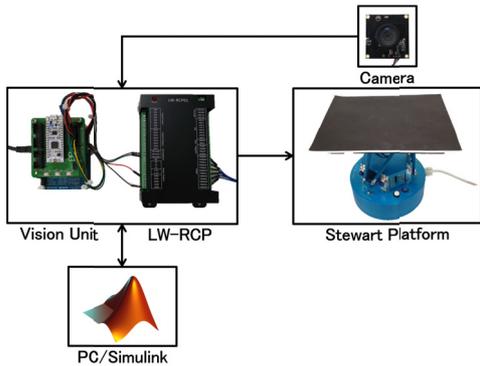


그림 4. 제안된 ball-and-plate 시스템의 하드웨어 배치도.
Fig. 4. Hardware layout of proposed ball-and-plate systems.

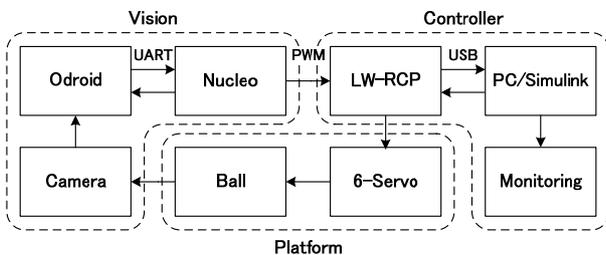


그림 5. 제안된 ball-and-plate 시스템의 구성도.
Fig. 5. Concept diagram of the proposed ball-and-plate system.

표 1. 센서 값 전달을 위한 UART 자체 통신규격.

Table 1. Custum UART protocol for transferring ball location.

Definition	Byte Size
Start flag	1 (=0x58)
X data	8
Change flag	1 (=0x59)
Y data	8

에 따른 움직임만을 가질 것을 가정하여 모델을 선정한다.

본 시스템을 사용한 실제 교육 시 사용 순서는 다음과 같다. 먼저 센서부를 구현하기 위해 학습자들은 머신비전 알고리즘을 설계한다. 알고리즘의 구현은 SBC의 운영체제 위에서 이루어지며 OpenCV를 이용하여 공 추적 알고리즘을 설계한다. 산출된 공의 위치는 시리얼 통신을 통해 마이크로컨트롤러로 다음 표 1을 따라 전송한다. 마이크로컨트롤러에서는 그 값을 PWM 형태로 변환하여 LW-RCP 시스템에 전송한다. 그리고 Matlab과 LW-RCP를 이용해 사용자들은 제어기를 설계하고 실시간 실험을 할 수 있다.

IV. 영상처리 및 공 탐색 알고리즘

1. 독립적 머신비전 시스템의 구성

공의 위치 탐색을 위한 독립적 머신비전 시스템은 SBC와 카메라 그리고 마이크로컨트롤러로 이루어진다. 실제 공의 위치를 알아내는 머신비전 알고리즘 연산은 ARM 프로세서 기반의 embedded I/O들을 갖는 SBC Odroid-XU4가 처리한다. Raspberry pi와 같은 다른 저가형 SBC와는 달리 Odroid-XU4는 병렬처리가 가능한 GPU인 Mali-T628가 있어 OpenCL을 이용한 실시간 머신비전 알고리즘 처리가 가능하다. 또한 운영체제 기반의 프로그램 동작이 가능하므로 OpenCV를 사용하여 학습자들은 더 쉽게 자신이 구상한 머신비전 알고리즘을 설계할 수 있다.

그러나 운영체제 위에서 처리되는 영상처리는 실시간성이 보장되지 않고 매 새로운 영상입력에 따라 영상처리에 소요되는 시간이 달라지는 문제가 있다. 제어시스템은 정확한 제어주기에 따른 실시간 연산을 요구하기 때문에 이처럼 불규칙한 처리 시간을 갖는 센서 값 갱신은 전체 제어 성능을 안 좋게 한다. 이를 방지하기 위해 본 논문에서 제안하는 머신비전 시스템은 제어기와 센서의 사이에서 정확한 시간 동기화가 가능하게 하는 연결 다리 역할의 마이크로컨트롤러를 장치한다. 그림 6과 같이 운영체제 기반의 영상처리가 21~23 ms 불규칙한 연산 지연시간을 가져도 중간의 마이크로컨트롤러가 제어주기에 따른 25 ms의 규칙적인 연산 지연시간을 갖도록 바꾼다. 이 역할을 위해 크기가 작은 오픈소스 하드웨어인 Nucleo-F303K8 보드를 사용한다.

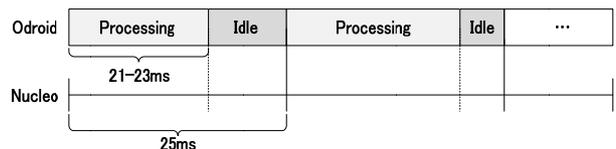


그림 6. 제안하는 머신비전 시스템의 시간 다이어그램.
Fig. 6. Timing diagram of the proposed machine vision system.

카메라는 전체 머신비전 알고리즘의 처리 비용을 줄이기 위해 VGA(640x480)의 작은 화면 크기를 갖고 회색조 형태의 영상을 취득하며 최대 90 fps 속도로 영상을 취득하면서도 셔터 속도에 의한 영상의 뭉개짐이 없는 global shutter 방식의 셔터 구조와 USB 웹캠의 데이터 출력이 가능한 KYT-U030-GS01을 사용한다.

2. 판 움직임에 강인한 머신비전 알고리즘

센서부에 적용되는 머신비전 알고리즘은 다음 두 가지 조건을 만족하며 설계된다. 첫째, 카메라 기반 시스템의 안정적 제어를 위하여 영상처리에 소요되는 시간은 충분한 제어주기 내에 완료되도록 설계한다. 둘째, 6-DOF의 구동부에 따라 다양한 움직임이 가능한 ball-and-plate 시스템을 가정하여 설계한다. Ball-and-plate 시스템에서 사용하는 대표적인 머신비전 알고리즘으로 히스토그램 기반의 추적알고리즘과 모양기반의 추적알고리즘이 있다[18]. 히스토그램 기반의 추적 알고리즘은 모양 기반의 적 알고리즘에 비해 노이즈가 많고 6-DOF로 움직이는 판의 위치도 알아낼 수 있어야 하므로 아래와 같은 모양기반의 머신비전 알고리즘을 사용한다.

공 탐색을 위한 머신비전 알고리즘은 그림 7과 같이 동작한다. 크게 제어주기 동기화, 판 추적, 공 위치 탐색, 결과 값 전송 네 가지 순서로 알고리즘을 요약할 수 있다. 먼저 Odroid 보드가 영상입력을 가져오기 전 제어기와의 시간동기를 위해 Nucleo 보드로부터 보내지는 시작 신호를 대기한다. Nucleo 보드로부터 신호를 받은 후 카메라로부터 회색조 영상을 그림 8의 (a)와 같이 받아온다. 그리고 공 탐색 알고리

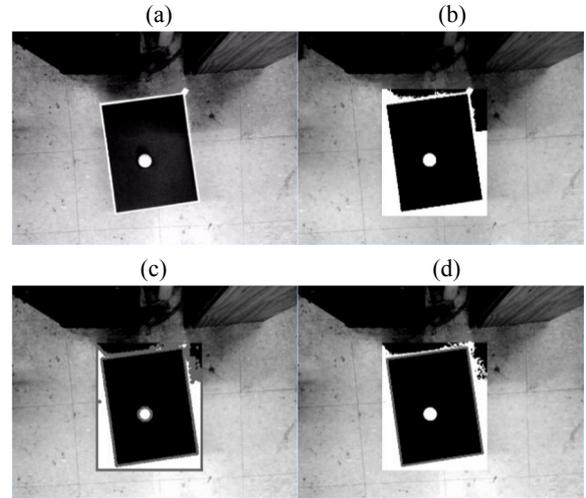


그림 8. (a) 카메라 입력 영상. (b)ROI영역 추출. (c)Contour 추출. (d) 판 추적.
 Fig. 8. (a) Captured frame. (b) Cropping ROI. (c) Extract contours. (d) Tracking plate.

즘의 비용을 줄이기 위한 ROI(Region Of Interest)를 이용한 판의 추적을 실시한다. 그림 8의 (b)와 같이 이전 판의 위치정보를 이용해 ROI를 설정하고 영상을 이진화한다.

이진화된 영상에서 판을 추적하기 위해 각진 도형들을 선별한다. 이를 위하여 지정된 ROI에 대해 그림 8의 (c)와 같이 contour들을 구한다. Contour는 영상 내 픽셀값이 같은 점들을 연결하여 주는 작업으로 연결된 점들에 대해 Ramer-Douglas-Peucker 알고리즘을 적용하여 도형화할 수 있다[19]. 도형화 작업을 거쳐 선별된 도형 중 공이 올라간 판을 찾아야 하므로 사각형이며 일정 이상의 크기를 갖는 도형을 그림 8의 (d)와 같이 판으로 선정한다.

판의 위치를 얻은 후 전체 화면에 대한 공의 위치를 알면 판을 기준으로 하는 공의 위치를 구할 수 있다. 그러나 공은 카메라 각도나 ball-and-plate의 기울인 각도 등에 의해 원의 형태로 존재하지 않을 수 있다. 이는 영상처리를 통한 공의 위치가 정확하지 않도록 만든다. 또 사용하는 구동부는 다양한 판의 움직임이 가능하기 때문에 새로운 영상입력에 대해 판의 위치가 바뀌어 전체 좌표계에서 공의 위치 탐색을 위한 연산이 길어진다. 영상처리에 소요되는 시간을 단축하기 위해 찾은 판의 위치를 기준으로 Homography 변환을 적용한다 [19]. 미리 정의된 판의 네 꼭짓점의 좌표와 전체 영상에서 탐색된 판의 네 꼭짓점 좌표를 식 (1)에 대입해 풀어주면 Homography 행렬을 구해줄 수 있다.

$$\begin{bmatrix} x_1 & y_1 & 0 & 0 & 1 & 0 \\ 0 & 0 & x_1 & y_1 & 0 & 1 \\ x_2 & y_2 & 0 & 0 & 1 & 0 \\ 0 & 0 & x_2 & y_2 & 0 & 1 \\ x_3 & y_3 & 0 & 0 & 1 & 0 \\ 0 & 0 & x_3 & y_3 & 0 & 1 \\ x_4 & y_4 & 0 & 0 & 1 & 0 \\ 0 & 0 & x_4 & y_4 & 0 & 1 \end{bmatrix} \begin{bmatrix} h_{11} \\ h_{12} \\ h_{13} \\ h_{21} \\ h_{22} \\ h_{23} \\ h_{31} \\ h_{32} \end{bmatrix} = \begin{bmatrix} x'_1 \\ y'_1 \\ x'_2 \\ y'_2 \\ x'_3 \\ y'_3 \\ x'_4 \\ y'_4 \end{bmatrix} \quad (1)$$

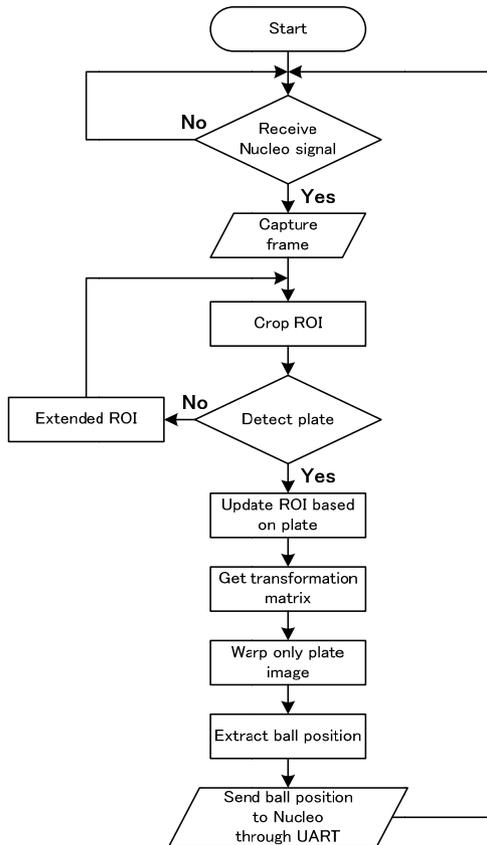


그림 7. 제안하는 머신비전 알고리즘 순서도.
 Fig. 7. Proposed machine vision algorithm flowchart.

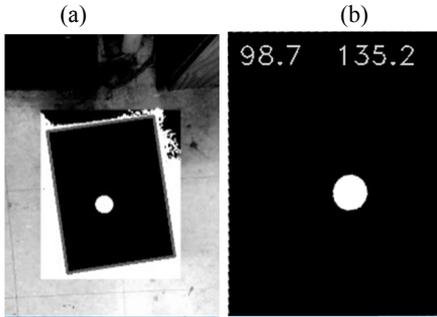


그림 9. (a) Homography 변환 전 관영상, (b) 변환 후 새로운 좌표계에 대한 관영상.

Fig. 9. (a) Plate image before homography transformation, (b) applied plate image for new coordinate.

여기서 $x_1, y_1 \sim x_4, y_4$ 는 그림 9의 (a)와 같은 전체 영상에서 구해진 판의 네 꼭짓점의 좌푯값이고 $x'_1, y'_1 \sim x'_4, y'_4$ 는 그림 9의 (b)와 같이 정의된 판의 네 꼭짓점 좌푯값이다. 구한 Homography를 식 (2)와 같이 전체 영상에서의 판이 차지하는 영역의 픽셀 좌푯값들을 넣어 연산하면 실제 판의 크기에 맞게 재구성한 영상을 얻을 수 있다. 공을 찾기 위해서 이진화 되어있는 재구성된 판 영상에 대해 OpenCV에서 제공하는 labeling algorithm을 적용하면 공의 위치를 알 수 있다[15]. 찾아낸 공의 위치 정보는 UART 통신을 통해 Nucleo보드로 전송한다.

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \quad (2)$$

V. 시스템 모델링과 LQR제어기 설계 및 구현

1. 제어부 하드웨어와 운동역학

제안하는 ball-and-plate 시스템에서 공의 위치를 제어하는 구동부로 병렬로봇의 일종인 Stewart platform이 사용된다. Stewart platform은 6-DOF를 가지며 작은 동작 범위 내에서 다양한 움직임을 만들어 낼 수 있어 다양한 제어 전략을 세워 공을 제어할 수 있게 해준다. 논문 [20]에서는 Stewart platform의 교육적 이점을 제시하며 저비용으로 간단히 구현할 수 있는 ball-and-plate 구동부를 제안했다. 본 논문에서는 이 구동

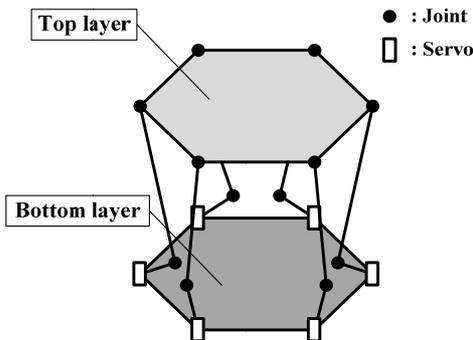


그림 10. Stewart platform의 구조.
Fig. 10. Structure of a Stewart platform.

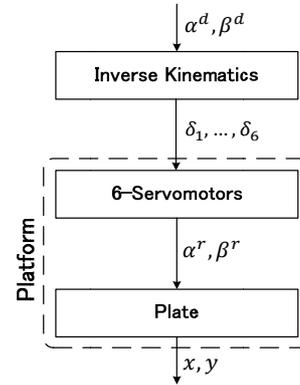


그림 11. Stewart platform 구동부의 판 회전 과정.
Fig. 11. Process of rotating the Stewart platform actuation.

부를 채택해 본 시스템의 구동부로 사용한다. 따라서 구동부의 운동역학과 선형화 모델 또한 [20]의 방법을 따른다.

제안된 Stewart platform이 적용된 ball-and-plate 시스템의 구동부는 그림 10과 같은 다이어그램으로 나타난다. Bottom layer에 조립된 6개의 서보모터의 동작에 따라 joint와 link로 연결된 top layer가 움직인다. Top layer의 움직임에 따라 함께 고정된 판이 기울어져 공의 위치가 변한다. 이때 사용된 6개의 서보모터의 성능은 6V 전압을 기준으로 0.08초에 60도까지 돌아간다. Stewart platform을 이용한 판의 회전과 그로 인한 공의 위치 변화는 그림 11과 같이 동작한다. 먼저 공을 제어하기 위해 제어기로부터 산출된 각도 정보 α^d, β^d 받아온다. 판을 그 기울기로 만들기 위해서 실제 실험에 사용하는 하드웨어에 따른 역운동역학 식을 적용해 각 서보모터가 움직여야 하는 각도 성분을 δ_1 부터 δ_6 까지 구한다[20]. 구한 각도 성분을 각 서보모터에 적용하면 실제 α^r, β^r 각도로 판이 기울여지고 그에 따라 공이 움직여 x, y 의 좌표에 위치하게 된다.

2. 모델링

그림 12는 ball-and-plate 시스템의 동작을 위한 개념도를 보여준다. α, β 는 각 축에 대해 기울어진 각도를 의미하며 x, y 는 공의 위치를 나타낸다. 위와 같은 구조의 ball-and-plate 시스템의 선형화된 모델은 중력가속도 g 를 포함하여 다음과 같이 표현할 수 있다[20].

$$\begin{aligned} \ddot{x} + \frac{5}{7}g\alpha &= 0, \\ \ddot{y} + \frac{5}{7}g\beta &= 0. \end{aligned} \quad (3)$$

위 식 (3)처럼 선형화된 식은 각 축에 대해 분리되어 있으므로 단일입력 단일출력 시스템이 두 개 있다고 볼 수 있다. 그래서 제어기 설계 또한 각 축에 대한 제어기를 따로 구성해 진행한다. 한 축에 대한 연속 시간 상태 방정식 모델은 식 (4)로 나타난다.

$$\begin{aligned} \dot{x} &= A_c x + B_c u, \\ y &= C_c x. \end{aligned} \quad (4)$$

여기서 A_c, B_c, C_c 그리고 x 는 다음과 같다.

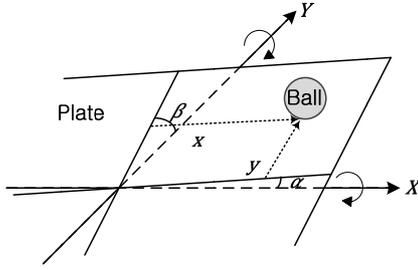


그림 12. Ball-and-plate 시스템 다이어그램.

Fig. 12. Diagram of the ball-and-plate system.

$$\begin{aligned} A_c &= \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}, B_c = \begin{bmatrix} 0 \\ -\frac{5}{7}g \end{bmatrix}, \\ C_c &= \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, x = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}. \end{aligned} \quad (5)$$

x_1 과 x_2 는 각각 공의 위치 정보와 속도 정보이다. 이론적으로 식 (5)의 모델을 이용해 제어기를 설계하면 공을 제어할 수 있어야 한다. 그러나 실제 실험환경에서는 바닥 면 기울기에 의한 offset 오차 때문에 공을 제어할 수 없다. 이를 해결하기 위해 본 논문에서는 공의 거리 오차에 대한 적분 요소를 추가한 식 (6)처럼 수정된 각 행렬을 모델방정식에 적용해 사용한다.

$$\begin{aligned} A_c &= \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix}, B_c = \begin{bmatrix} 0 \\ -\frac{5}{7}g \\ 0 \end{bmatrix}, \\ C_c &= \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, x = \begin{bmatrix} x_1 \\ x_2 \\ \int x_1 \end{bmatrix}. \end{aligned} \quad (6)$$

위 식 (6)의 연속 시간 상태 방정식을 실제 제어기에 적용하기 위해 이산 시간 상태 방정식으로 식 (7)과 같이 표현한다. 이산화에 따른 각 A_d, B_d, C_d 행렬은 Matlab 함수 c2d를 이용해 구할 수 있다.

$$\begin{aligned} x(k+1) &= A_d x(k) + B_d u(k), \\ y(k) &= C_d x(k). \end{aligned} \quad (7)$$

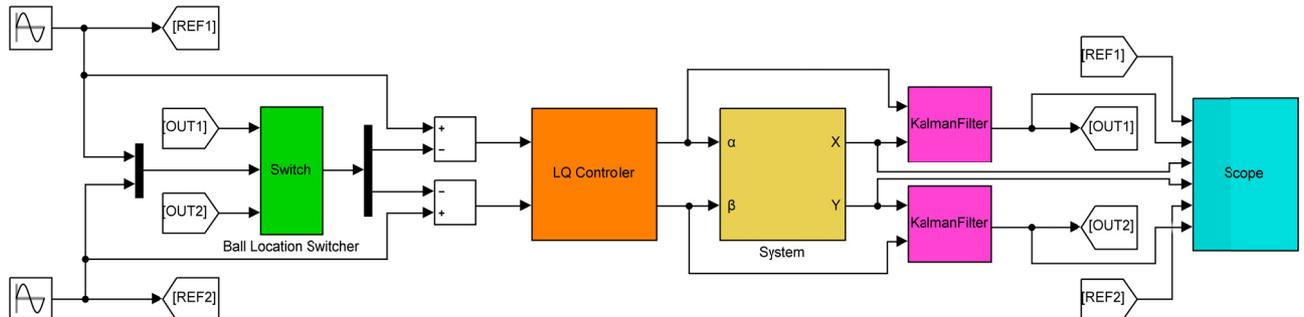


그림 13. 제안하는 ball-and-plate 시스템의 전체 제어 블록 선도.

Fig. 13. Control block diagram for a proposed ball-and-plate system.

3. LW-RCP에서의 LQ모델 적용

식 (7)에 구한 모델방정식을 이용해 ball-and-plate 시스템에 대한 최적화된 LQ 제어기를 설계할 수 있다. 제어기의 목표는 공의 상태와 입력 각도를 최소화하는 입력을 찾는 것이다. 비용 함수 J 는 가중행렬 Q, R 과 함께 식 (8)과 같이 나타난다.

$$J_\infty = \sum_{i=1}^{\infty} [x^T(k+i)Qx(k+i) + u^T(k+i)Ru(k+i)] \quad (8)$$

비용 함수 J 를 최소화하는 입력 u 는 식 (9)처럼 현재 상태 $x(k)$ 과 원하는 기준 상태 $x_r(k)$ 의 차이에 LQ이득 계수인 K 를 곱한 형태로 표현된다. 제어기 입력을 만들기 위해 K 를 구해야 하고 K 를 구성하는 행렬 P 는 이산 시간에 대한 Riccati 방정식을 사용해 구한다. Matlab 환경에서 LQ제어기를 구현하기 위한 되먹임 이득 행렬 K 는 dlqr 함수를 사용해 구할 수 있다.

$$\begin{aligned} u(k) &= -K[x(k) - x_r(k)], \\ K &= (B_d^T P B_d + R)^{-1} B_d^T P A_d, \\ P &= A_d^T P A_d - A_d^T P B_d (R + B_d^T P B_d)^{-1} B_d^T P A_d + Q. \end{aligned} \quad (9)$$

제안하는 시스템은 LW-RCP를 사용해 PC의 Matlab-Simulink 환경에서 제어기 설계와 실시간 제어 실험이 가능하다. Simulink에서 전체 ball-and-plate 시스템을 제어하기 위한 LQ 제어기가 적용된 블록선도는 그림 13과 같다. System 블록의 내부는 LW-RCP의 입출력 블록들을 사용하여 그림 14와 같이 구성된다. 'Period/Duty' 블록을 이용해 센서부로부터 보내지는 Duty 값을 받아오며 제어기에 사용되는 좌표계에 맞는 값으로 변환한다. 또 제어연산의 결과로 만들어진 각 좌표축에 필요한 각도 값을 역운동역학 블록을 거쳐 'PWM' 블록을 통해 구동부로 출력한다.

VI. 실험 및 결과

제안하는 시스템의 유용성을 확인하기 위해 다음 두 실험을 실시한다. 먼저 센서부에 적용한 머신비전 알고리즘을 검증하고 검증된 센서에서 측정되는 측정 값을 통해 설계한 제어기를 검증한다. 제어기 검증은 원점에 대한 균형제어와 원 형태의 경로 쫓는 추종제어를 실시한다. 실험 환경으로 공이

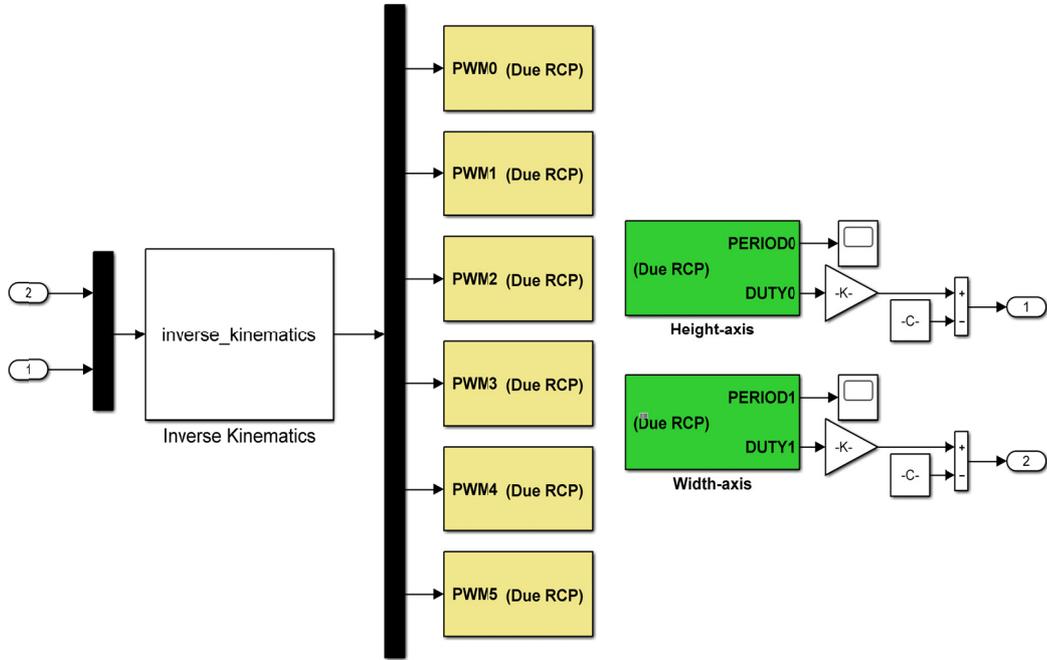


그림 14. LW-RCP의 I/O블록을 사용한 시스템 블록의 설계.
Fig. 14. Design of system block using I/O blocks from LW-RCP.

위치한 직사각형 판의 크기는 200x240 mm이며 카메라는 판으로 부터 400 mm 높이에 위치한다. 또 머신비전 알고리즘 검증에 위해 연구실에서 자체 제작한 테스트베드는 원판의 형태를 가지며 공이 지날 수 있는 70 mm의 원형 경로를 갖는다.

먼저 센서부의 검증을 위해 테스트 베드를 사용한 공의 위치 측정을 실시한다. 개루프 제어를 통해 공이 원을 돌 수 있도록 각 축 x, y에서 sin과 cos과 형태의 기울기를 만든다. 만들어진 기울기는 실시간으로 역운동역학식을 통해 구동부를 움직이는 신호를 만들어 공을 움직이게 한다. 센서부에서 측정된 공의 위치와 70 mm의 반지름의 참조 값은 그림 15의 (a)와 같이 나타난다. 센서부의 측정 값이 어느 특정 방향에 대해 기울여짐이나 offset 없이 동작함을 확인할 수 있다. 또 각 각도에 따른 측정 값과 참조 값과의 오차는 그림 15의 (b) 처럼 나타난다. 최대 오차는 약 1 mm이며 적은 노이즈가 포함된다. 이 센서 값을 제어기 검증에 사용하기 위해 칼만필터를 적용하였다.

제어기 검증을 위한 첫 번째 제어목표로 원점에 대한 균형 제어를 실시한다. 공은 원점으로부터 떨어진 곳에서 시작하며 판은 기울여지지 않은 평평한 상태로 시작된다. 결과는 시간에 따른 위치 오차로서 그림 16의 두 그래프처럼 나타난다. 그림 16의 (a)는 X좌표축에 대한 공의 오차 그래프, 그림 16의 (b)는 Y좌표축에 대한 공의 오차 그래프이다. 약 4초에 걸쳐 공은 원점으로 수렴함을 보여 균형제어가 가능함을 볼 수 있다.

두 번째 제어목표로 주어진 참조 경로를 쫓는 추종제어를 실시한다. 참조 경로는 반지름 40 mm의 원형 경로로 주어지며 8초에 거쳐 한 바퀴를 도는 속도를 갖는다. 제어기를 통한 공의 움직임은 그림 17의 (a)와 같이 나온다. 각 축에 대

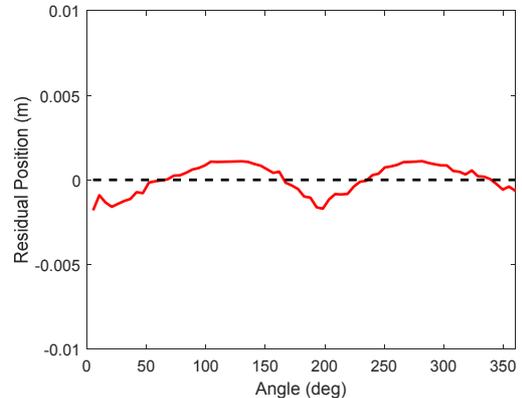
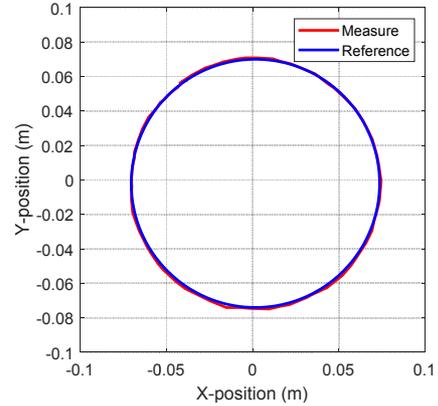


그림 15. (a) 테스트베드를 따라 움직인 공의 궤적 측정, (b) 각도에 따른 기준 원과의 거리 차이.
Fig. 15. (a) The ball tracking measurement along the testbed, (b) the difference between reference circle and measurement depending on the angle.

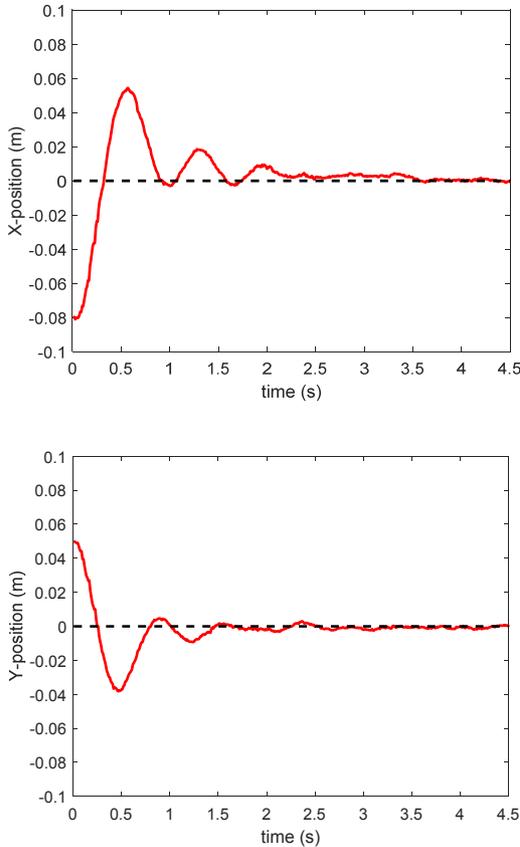


그림 16. 제안하는 시스템의 균형제어 결과 오차. (a) X축 기준, (b) Y축 기준.
 Fig. 16. Error in balance control result of proposed system. (a) x-axis reference, (b) y-axis reference.

한 오차는 그림 17의 (b)와 같이 나오며 주어진 원형 경로에 대해 최대 16mm의 오차를 갖는다. 본 실험은 머신비전 알고리즘 처리 시간에 맞춰 25ms로 제어주기를 설정했다. 많은 영상을 기반으로 하는 ball-and-plate 시스템 논문의 제어 실험의 결과는 영상처리가 빠를수록 제어 성능이 향상됨을 보인다. 논문 [14]에서는 영상을 기반으로 하는 ball-and-plate 시스템을 이용해 PID 제어 실험을 보였는데 단순한 시뮬레이션에서도 43ms 이상의 제어주기를 갖는다면 시스템이 제어되지 않음을 보였다. 본 시스템을 이용한 제어 실험에서는 빠른 제어주기를 얻기 위해 주어진 하드웨어에 최적화된 머신비전 알고리즘 설계를 해서 25 ms를 제어주기로 동작했다. 이는 논문[15]에서 보인 Raspberry Pi2를 사용한 66ms의 동작주기보다 빠르다. 결과로서 공의 제어까지 걸리는 시간은 앞선논문에서 28s가 걸렸다면 본 논문에서는 4s의 시간으로 공의 완전한 제가 이루어짐을 확인할 수 있다. 그러나 영상기반 ball-and-plate 시스템이 정상적인 제어동작을 보이기 위해서는 더 빠른 영상처리 속도와 그러면서도 정확한 공의 위치 측정 알고리즘이 적용되어야 한다.

VII. 결론

본 논문에서는 연구실에서 만든 ball-and-plate 시스템과 LW-RCP, 그리고 SBC기반의 머신비전 시스템으로 구성된 교

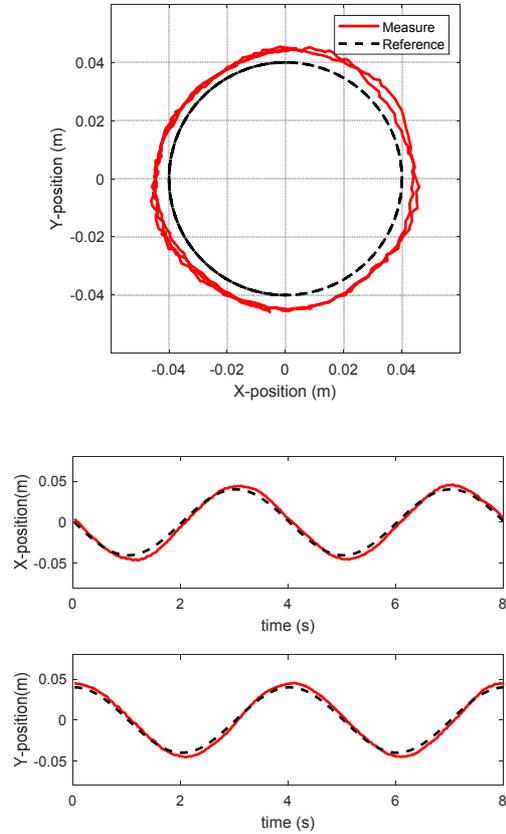


그림 17. (a) 제안하는 시스템의 추종제어 결과, (b) 각 축에 대한 위치 오차.
 Fig. 17. (a) Control output results of the proposed system, (b) the position error of each axis.

육적 목적의 실습도구로 사용 가능한 ball-and-plate 시스템 구조를 제안하였다. 제어공학 로봇공학, 머신비전 등의 학습을 실험해볼 수 있는 본 시스템은 특정 제어 문제를 설정하고 그것을 해결하기 위한 제어기 설계, 머신비전 알고리즘 설계를 보이고 실제 실험을 통해 제안하는 시스템이 안정적으로 동작함을 보였다.

기존 제안된 영상기반 시스템들보다 마이크로컨트롤러와 SBC 기반의 정확한 머신비전 시스템의 구성은 구성하기 위한 가격 대비 뛰어난 제어 성능을 보이고 LW-RCP기반의 빠른 설계와 실시간 제어실험이 가능해 큰 교육적 효과를 가진다. Stewart platform의 6-DOF 움직임을 이용해 다양한 제어 목표를 설정할 수 있으며 학습자들이 원하는 분야의 학습만을 선택적으로 학습할 수 있다.

REFERENCES

- [1] H. Jee, "Industry joint engineering education via interdisciplinary team-based product development project," *Journal of Engineering Education Research*, vol. 16, no. 3, pp. 51-60, 2013.
- [2] A. Leva, "A hands-on experimental laboratory for undergraduate courses in automatic control," *IEEE Transactions on Education*, vol. 46, no. 2, pp. 263-272, May 2003.
- [3] L. D. Feisel and J. R. Albert, "The role of the laboratory in undergraduate engineering education," *Journal of Engineering Education*, vol. 94, no. 1, pp. 121-130, January 2005.

- [4] M. Gunasekaran and P. Ramprasa, "Low-cost undergraduate control systems experiments using microcontroller-based control of a DC motor," *IEEE Transactions on Education*, vol. 55, no. 4, pp. 508-516, Apr. 2012.
- [5] Y. S. Lee, J. J. Oh, S. Y. Shim, H. Lim, and J. H. Seo, "Swing-up control for a rotary inverted pendulum with re-restricted rotation range," *Journal of Institute of Control, Robotics and Systems (in Korean)*, vol. 20, pp. 835-841, 2014.
- [6] Y.-S. Oh and Y. S. Lee, "Robust swing-up control of a rotary inverted pendulum subject to input/output constraints," *Journal of Institute of Control, Robotics and Systems (in Korean)*, vol. 24, no. 5, pp. 423-430, 2018.
- [7] C. Lee and Y. S. Lee, "Implementation of 2-DOF inverted pendulum control system robust against actuator uncertainties," *Journal of Institute of Control, Robotics and Systems (in Korean)*, vol. 24, no. 5, pp. 377-385, 2018.
- [8] C. J. Bay and B. P. Rasmussen, "Exploring controls education: A re-configurable ball and plate platform kit," *Proc. Of 2016 American Control Conference (ACC)*, pp. 6652-6657, Jul. 2016.
- [9] J. Hauser, S. Sastry, and P. Kokotovic, "Nonlinear control via approximate input-output linearization: The ball and beam example," *IEEE transactions on automatic control*, vol. 37, no. 3, pp. 392-398, 1992.
- [10] S. Awatar, C. Bernard, N. Boklund, A. Master, D. Ueda, and K. Craig, "Mechatronic design of a ball-on-plate balancing system," *Mechatronics*, vol. 12, no. 2, pp. 217-228, 2002.
- [11] J. H. Park and Y. J. Lee, "Robust visual servoing for motion control of the ball on a plate," *Mechatronics*, vol. 13, no. 7, pp. 723-738, 2003.
- [12] D. Stander, S. Jiménez-Leudo, and N. Quijano, "Low-Cost "ball and Plate" design and implementation for learning control systems," *Proc. of 2017 IEEE 3rd Colombian Conference on Automatic Control (CCAC)*, Cartagena, Colombia, Feb. 2018.
- [13] J. M. Rossell, J. Vicente-Rodrigo, J. Rúbio-Massegú, and V. Barcons, "An effective strategy of real-time vision-based control for a Stewart platform," *Proc. of 2018 IEEE International Conference on Industrial Technology (ICIT)*, Lyon, France, Apr. 2018.
- [14] M. -T. Ho, Y. Rizal, and L. -M. Chu, "Visual servoing tracking control of a ball and plate system: Design, implementation and experimental validation," *International Journal of Advanced Robotic Systems*, vol. 10, no. 7, Jan. 2013.
- [15] E. Ali and N. Aphiratsakun, "AU ball on plate balancing robot," *Proc. Of 2015 IEEE International Conference on Robotics and Biomimetics (ROBIO)*, pp. 2031-2034, Dec. 2015.
- [16] H. Cheng, L. Hao, Z. Luo, and F. Wang, "Establishing the connection between control theory education and application: An arduino based rapid control prototyping approach," *International Journal of Learning and Teaching*, vol. 2, no. 1, pp. 67-72, Jun. 2016.
- [17] Y. S. Lee, B. Jo, and S. Han, "A light-weight rapid control prototyping system based on open source hardware," *IEEE Access*, vol. 5, pp. 11118-11130, Jun. 2017.
- [18] Y.-K. Park, J. -Y. Park, and S. -M. Park, "Design of ball and plate robot controller using single camera," *Journal of Korea Multimedia Society*, vol. 16, no. 2, pp. 213-225, Feb. 2013.
- [19] S. Richard, *Computer vision: algorithms and applications*, 1st ed., Springer-Verlag, New York, 2010.

- [20] H. Bang and Y. S. Lee, "Implementation of a ball and plate control system using sliding mode control," *IEEE Access*, vol. 6, pp. 32401-32408, May 2018.



박주현

2018년 인하대학교 정보통신공학과 졸업. 2018년~현재 인하대학교 대학원 전기공학과 석사과정 재학중. 관심분야는 임베디드 시스템, 로봇 공학, 머신비전, 객체 추적.



방희승

2018년 인하대학교 기계공학과 졸업. 2020년 동 대학원 전기공학과 석사. 관심분야는 로봇공학, 최적제어, 임베디드 시스템.

이영삼

제어로봇시스템 학회 논문지, 제 15 권 제 4 호 참조.