

안드로이드 스마트폰 기반 이동형 피칭 머신의 설계 및 구현

Design and Implementation of the Pitching Machine with Mobility using the Android Smartphone

박성용* · 오경현* · 최호림†
(Sung-Yong Park · Kyung-Hyun Oh · Ho-Lim Choi)

Abstract - Pitching machines have been around for many years for casual amusement purpose or professional athletes' training usage, and so forth. The current pitching machines are usually built on the firm ground and do not have any mobility function. In this paper, we develop a pitching machine that has both ball-shooting and mobility functions. Our developed pitching machine consists of two parts. The upper body part has a function of shooting a ball using two DC motors and the lower body part has a function of mobility like mobile robots by using two wheels governed by DC motors. All these functions are operated wirelessly by Android smartphones via bluetooth. The control of each DC motor is done by ϵ -PID control method in which the gain tuning is simplified by using a single parameter ϵ . Simulation and actual experiment show that our developed pitching machine has both nontrivial shooting and mobility features.

Key Words : Pitching machine, Mobility, Android, Smartphone, Bluetooth, Remote control

1. 서론

현재 피칭머신의 특성은 모터제어나 움직임, 공 충전 방식등을 수동으로 조작해야 작동을 한다는 것이다. 이러한 특성을 개선시키기 위해 스마트폰을 이용하여 좀 더 능동적인 형태를 가지게 하려고 한다. 또한 스마트폰은 가속도 센서, GPS 등의 각종 센서가 적용이 되고 있고, 블루투스 및 Wi-Fi와 데이터 처리를 위한 CPU의 성능이 크게 향상되고 있다. 이를 이용한 다양한 산업분야에서 로봇이나 자동차 등의 기계와 스마트폰을 연동하려는 다양한 연구들이 진행되고 있다[8]. 아두이노는 IoT(Internet of Things)기반의 개방형 플랫폼이다. 이를 이용하여 스마트폰과 블루투스를 연동하여 제어 할 수 있다. IoT란 사물에 센서를 부착해 실시간으로 데이터를 모아 인터넷으로 전달해주는 것이며, 이는 사물간 데이터를 주고받은 후 이를 다시 활용하는 프로세서라고 할 수 있다. 전자 기관의 하드웨어와 이를 제어하는 소프트웨어로 구성되어 있으며 개발을 위한 도구로 이용되기에 오픈 소스(Open source)형태이고, 이는 쉬운 접근성과 장비 구성 및 소스 작성에 유연성을 가진다. 이는 기본적인 하드웨어 지식을 익힌다면 윈도우, MAC, 리눅스OS 등등 다양한 OS에서도 활용 할 수 있다는 장점으로 부각된다[7]. 앞에서 말했듯이 기존의 피칭머신의 수동적인 특성을 개선하기위해 본 논문을 통해 발사부분의 모터와 하단부의 움직임을 스마트폰을 이용하여 제어하고 기존의 특성을 최소화시켜 본 논문에서 제시된 피칭머신의 장점으로 나타낸다.

본 논문은 다음과 같이 구성되어 있다. 2장에서는 피칭머신의 개요 및 세부 구조를 나타내어 블루투스통신이 어떻게 적용 되었는지와 제어 방식을 설명하고 3장은 모터의 운동 방정식과 상단부와 하단부로 나뉘어져 ϵ -PID 제어 방식이 나오며 4장은 시뮬레이션 및 실험 결과를 보여 준다.

2. 피칭 머신의 개요 및 세부 구조

2.1 피칭 머신의 개요

그림 1은 본 논문의 실제 피칭머신 개요도이며 그림 2는 실제로 제작된 피칭머신이다. 크게 상단부와 하단부로 나뉘어져 있으며 스마트폰과 피칭머신과의 블루투스 통신을 이

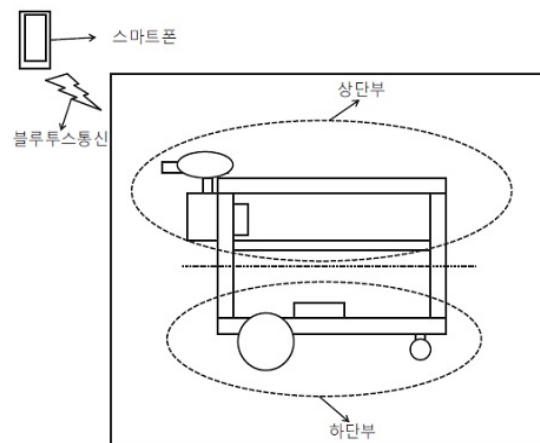


그림 1 피칭 머신의 전체개요도
Fig. 1 Design schematic of pitching machine

* Dept. of Electrical Engineering, Dong-A University, Korea
† Corresponding Author : Dept. of Electrical Engineering, Dong-A University, Korea

E-mail : hlchoi@dau.ac.kr

Received : May 7, 2014; Accepted : June 25, 2014

용하여 피칭 머신을 제어하게 된다. 피칭머신을 원거리에서 스마트폰으로 제어하기 위해서는 블루투스나 Wi-Fi같은 무선 통신을 이용하여 데이터를 전송할 수 있다. 본 논문에서 Wi-Fi가 아닌 블루투스를 이용하는 이유는 블루투스가 통신거리에는 제한이 있지만 단가 측면과 개인이 스마트폰에 어플리케이션을 깔거나 노트북을 이용하여 쉽고 편리하게 모니터링을 할 수 있다는 장점을 가지고 있어 사용하게 되었다[5]. 블루투스는 마스터와 슬레이브의 역할을 조절함으로써 전송거리를 확장할 수도 있다. 블루투스의 연결을 성립하기 위해서는 요청과 연결의 과정을 통해 패킷을 교환함으로써 정보를 교환한다[3]. 이러한 과정을 통해 스마트폰과 피칭머신의 블루투스 통신을 이용하여 상단부 모터의 RPM과 하단부 움직임에 위한 모터의 RPM을 제어한다.

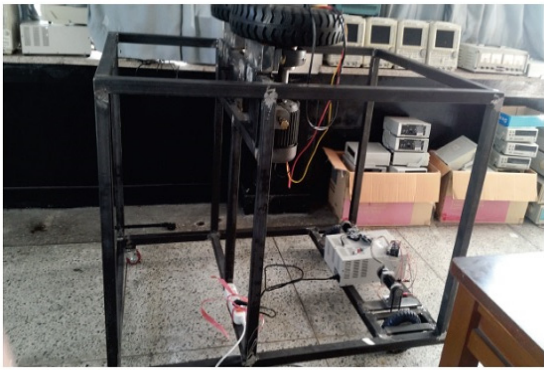


그림 2 제작된 피칭 머신
Fig. 2 Manufactured pitching machine

2.2 피칭머신의 세부 구조

그림 3은 피칭머신의 발사부 구조를 나타낸 것이다. 2개의 휠과 모터는 공을 발사하는 구동체로 작동한다. 모터 컨트롤러를 통해 모터의 rpm을 올리게 되면 휠의 각속도가 증가하게 된다. 이때 공을 넣게 되면 양쪽 휠의 구동으로 인해 휠의 진행방향대로 발사하게 된다. 이때 가이드바로 인해 공이 원하는 곳으로 더욱 안정하게 발사한다. 오른쪽 휠의 속도를 낮추게 되면 공은 회전력으로 인해 오른쪽으로 휘어지게 되며 반대로 왼쪽 휠의 속도를 낮추게 되면 왼쪽으로 휘어지게 된다.

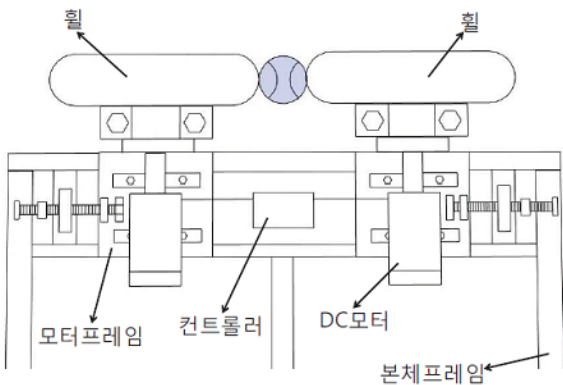


그림 3 상단부
Fig. 3 Upper body part

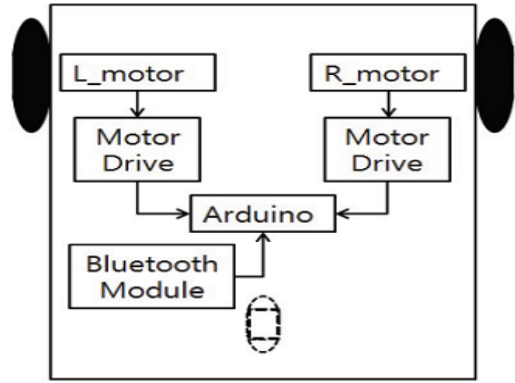


그림 4 하단부
Fig. 4 Lower body part

그림 4는 본 논문에서 제시된 피칭머신의 하단부이다. 사용자의 필요에 의해 움직일 수 있도록 제작되었다. 블루투스 모듈을 이용해 스마트폰의 블루투스와 테더링을 하여 모터 드라이브를 통해 각각의 모터를 제어 하도록 되어있다. 하단부는 상단부의 무게를 안정하게 견디며 작동이 가능해야 되기 때문에 속도의 변화를 주는 제어 보다는 일정 속도를 유지할 수 있도록 제어시스템을 설계하였다.

3. 모터 제어 및 설계

3.1 DC모터 운동 방정식

다음 식은 본 논문의 피칭머신에 쓰인 DC모터의 방정식이다[1].

$$V(t) = L_a \frac{di(t)}{dt} + R_a i(t) + e(t) \tag{1}$$

$$T_e = k_T \phi_f + i(t) \tag{2}$$

$$e(t) = k_e \phi_f \omega_m \tag{3}$$

$$T_e = J \frac{d\omega_m}{dt} + B\omega_m + T_L, \quad (J = J_m + J_L) \tag{4}$$

$V(t)$ 는 전기자에 가해지는 전압, $i(t)$ 는 전기자 권선의 전류, R_a 는 전기자 권선의 저항, L_a 는 전기자 권선의 인덕턴스, $e(t)$ 는 역기전력이다. 식 (1)은 전기자 회로의 전압식이며, 식 (2)는 토크, 식 (3)은 유도 기전력, 식 (4)는 부하의 운동 식이다. k_e 는 역기전력 상수, ϕ_f 는 계자 자속, ω_m 은 회전자의 회전 각속도, B 는 점성마찰계수, k_T 는 토크상수이다. 식 (1), (2), (3) 그리고, (4)와 $\dot{\theta} = \omega_m$ 을 정리하여 상태 방정식을 구하면 식 (5), (6)으로 나타내진다.

$$\dot{x}_1 = x_2 \tag{5}$$

$$\dot{x}_2 = -\frac{R_a B + k_T k_e \phi_f^2}{L_a B + R_a J} x_2 + \frac{k_T \phi_f}{L_a B + R_a J} u - \frac{R_a}{L_a B + R_a J} T_L \tag{6}$$

$$x_1 = \theta, x_2 = \omega_m, V(t) = u \text{ 이고, } \frac{R_a B + k_T k_e \phi_f^2}{L_a B + R_a J} = a, \frac{k_T \phi_f}{L_a B + R_a J} = b,$$

$\frac{R_a}{L_a B + R_a J} = c$ 로 정리해서 사용한다.

그리고 다음과 같은 식으로 정리하여 사용할 수 있다.

$$\dot{x}_1 = x_2 \tag{7}$$

$$\dot{x}_2 = -ax_2 + bu - cT_L \tag{8}$$

식 (7), (8)의 상태방정식에서 각속도인 $x_1 (= \omega_m)$ 을 제어하기 위해서 ϵ -PID제어 방식을 사용한다[9].

$$e_1 = x_1 - r, \quad e_2 = x_2 \tag{9}$$

$$\dot{e}_1 = \dot{x}_1 - \dot{r} = x_1 - x_2$$

$$e_2 = x_2 = -ax_2 + bu - cT_L$$

부하토크인 T_L 의 측정은 고가의 장비가 필요하기 때문에 직접적으로 측정이 곤란하다. 그러므로 T_L 은 알 수 없다고 가정하는 경우가 많다[1]. 본 논문에서는 T_L 을 미분 가능하며, 한정된 상수를 가진 ρ 로 가정하고 $\dot{T}_L \leq \rho$ 라는 조건을 설정한다. 식 (9)를 미분하면 식 (10)과 같으며,

$$\dot{e}_1 = e_2 \tag{10}$$

$$\dot{e}_2 = -ae_2 + b\dot{u} - c\dot{T}_L$$

$\dot{u} = v$ 로 $z = [z_1, z_2, z_3]^T = [e_1, e_2, e_1]^T$ 로 두고 식 (10)과 합쳐 나타내면 식 (11)이 된다.

$$\dot{z} = Az + B(-az_2 + bv - c\dot{T}_L), \quad A = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} \tag{11}$$

폐환 선형화 제어기 식 (12)를

$$v = \frac{az_2 + \bar{v}}{b} \tag{12}$$

식 (11)에 적용하면 다음을 얻을 수 있다.

$$\dot{z} = Az + B(\bar{v} - c\dot{T}_L) \tag{13}$$

내부 제어기인 \bar{v} 를 식 (14)로 정의한다.

$$\bar{v} = \frac{k_1}{\epsilon^2} z_1 + \frac{k_2}{\epsilon} + \frac{k_3}{\epsilon^3} z_3 = \epsilon^{-2} KE_\epsilon z \tag{14}$$

여기서 $K = [k_1, k_2, k_3]$, $E_\epsilon = \text{diag}[1, \epsilon, \epsilon^{-1}]$, $\epsilon > 0$ 로 정하고 식 (14)를 식 (12)에 대입하면 다음과 같다.

$$v = \frac{k_1}{b\epsilon^2} z_1 + \frac{1}{b} \left(\frac{k_2}{\epsilon} + a \right) z_2 + \frac{k_3}{b\epsilon^3} z_3 \tag{15}$$

$\dot{u} = v$ 로 인하여 $e = e_1$ 으로 놔두고 ϵ -PID제어기를 최종 정리하면 다음과 같다.

$$u = K_P(\epsilon)e + K_D(\epsilon)\dot{e} + K_I(\epsilon) \int_0^t e(\tau) d\tau \tag{16}$$

PID의 이득값은 다음과 같다.

$$K_P(\epsilon) = \frac{k_1}{b\epsilon^2}, \quad K_D(\epsilon) = \frac{1}{b} \left(\frac{k_2}{\epsilon} + a \right), \quad K_I(\epsilon) = \frac{k_3}{b\epsilon^3} \tag{17}$$

이득값 k_1, k_2, k_3 와 단일 파라미터 ϵ 를 이용하여 정리 하는 것을 다음장에 설명 하도록 한다.

3.2 제어기의 안정성 및 성능 분석

제안된 ϵ -PID제어기에 대한 다음과 같은 설계과정을 제시한다.

(i) K 값을 정하기 위해서는 $A_K = A + BK$ 가 Hurwitz가 되어야 한다.

(ii) Lyapunov 방정식에 의해 $A_K^T P + PA_K = -I_n$ 을 만족하는 P 를 구한다.

(iii) ϵ -PID의 이득값 튜닝에서 ϵ 값을 정하기 위해 $\epsilon^{-1} - 2r_1 > 0$ 의 조건을 만족해야 하고 $r_1 = 2\|P\|$ 이다.

위 설계과정 (i)과 같은 K 값을 구하려면 모든근이 아래의 다항식에서 음의 실수를 가져야 한다.

$$0 = s^3 - k_2 s^2 - k_1 s - k_3 \tag{18}$$

정리를 하자면 A_K 가 Hurwitz이면 $A_K(\epsilon) := A + B\epsilon^{-2}KE_\epsilon$ 도 $\epsilon > 0$ 일 때 Hurwitz가 된다. 증명하자면 A_K 와 $A_K(\epsilon)$ 은 다음과 같은 관계가 성립된다[9].

$$A_K = \epsilon E_\epsilon A_K(\epsilon) E_\epsilon^{-1} \tag{19}$$

참고로 $\lambda(E_\epsilon A_K(\epsilon) E_\epsilon^{-1}) = \lambda(\epsilon^{-1} A_K)$ 와 $\lambda(\epsilon^{-1} A_K) = \epsilon^{-1} \lambda(A_K)$ 이고, $\lambda(M)$ 은 M 행렬의 고유값을 의미하며, A_K 가 Hurwitz일 때 $\epsilon > 0$ 이면 $A_K(\epsilon)$ 또한 Hurwitz임을 보장한다. 다시 언급하면, K 값과 ϵ 을 위에서 제시된 설계과정에 의해 결정되었다고 하고 식 (13)을 페루프시스템으로 정리하면 식 (18)이 되고,

$$\dot{z} = A_K z - Bc\dot{T}_L \tag{20}$$

A_K 는 Hurwitz이므로 다음의 Lyapunov방정식이 존재하는데

$$A_K^T P + PA_K = -I_n \tag{21}$$

식 (18)을 식 (21)에 적용하여 새로운 Lyapunov방정식으로 유도하면 다음과 같다.

$$A_K^T(\epsilon) P_\epsilon + P_\epsilon A_K(\epsilon) = -\epsilon^{-1} E_\epsilon^2, \quad P_\epsilon = E_\epsilon P E_\epsilon \tag{22}$$

식 (22)에서 얻은 Lyapunov방정식은 $A_K(\epsilon)$ 이 Hurwitz가 되고 P_ϵ 이 양수로 정의되면 시스템에서 유효성을 가진다. 이제 Lyapunov함수를 $V(z) = z^T P_\epsilon z$ 로 정의한다. 그런 다음 식 (20)의 궤도에 따라 시간에 대해 미분을 하면

$$\dot{V}(z) = -\epsilon^{-1} \|E_\epsilon z\|^2 + 2z^T E_\epsilon P_\epsilon E_\epsilon B(-c\dot{T}_L) \tag{23}$$

식 (23)으로부터 $E_\epsilon B(-c\dot{T}_L) \leq \|E_\epsilon\| + c\rho$ 인 조건하에, 식 (24)로 나타낼 수 있다.

$$\begin{aligned} \dot{V}(z) &\leq -\epsilon^{-1} \|E_\epsilon z\|^2 + r_1 \|e_\epsilon z\| \\ &= -\left(\frac{1}{2}\epsilon^{-1} - r_1\right) \|E_\epsilon z\|^2 - \epsilon^{-1} \left(\frac{1}{2}\|E_\epsilon z\| - \epsilon^2 r_2\right) \|E_\epsilon z\| \end{aligned} \tag{24}$$

여기서 $r_1 = 2\|P\|$, $r_2 = 2c\rho\|P\|$ 이다. ϵ 은 $\epsilon^{-1} - 2r_1 > 0$ 조건에 맞도록 설정하며 $\|E_\epsilon z\|$ 은 $O(\epsilon^2)$ 으로 묶는다. E_ϵ 으로부터 $\|z\|$ 은 $O(\epsilon)$ 으로 나타낼 수 있다. 그리고 부하토크인 T_L 은 상수인 경우 $\rho = 0$ 이다. 단일 파라미터인 ϵ 값을 바꾸면서 이득값을 결정하여 나타낸 시뮬레이션은 4.1장에서 나타내도록 한다.

3.3 상단부 모터 컨트롤러와 스마트폰 연동

그림 5(우)는 본 논문의 상단부 컨트롤러이다. 내부 볼륨을 이용하여 스마트폰과 연동하여 조작하거나, 외부의 단추나 볼륨을 이용하여 수동조작도 가능하도록 되어있다, LCD 기판을 이용해 모터의 RPM도 출력이 되도록 설정이 되어 있다. 상단부 모터(TM90-D0331)의 RPM제어는 BluCar 어플리케이션과 컨트롤러의 내부 볼륨에 아두이노와 블루투스 모듈을 적용시켜 스마트폰과 테더링 후 제어를 하게 된다.

테더링이 된 후에는 소스코드인 타이머를 이용하여 Forward나 Reverse버튼을 누르게 되면 누른 시간을 적용하여 휠의 RPM을 제어하게 된다. 이 때 컨트롤러에 있는

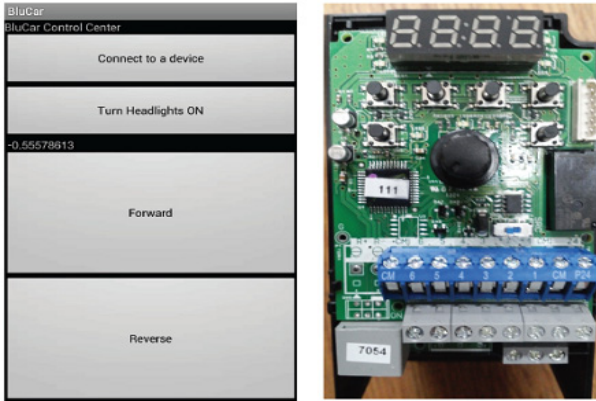


그림 5 BluCar어플리케이션(좌), 상단부 컨트롤러(우)
 Fig. 5 Blucar application(left), upper body part controller(right)

LCD기판에 RPM이 출력이 되도록 설정이 되어 있어 모터가 구동이 되었을 때 RPM을 알 수 있다. 그리고 스마트폰을 오른쪽이나 왼쪽으로 기울여 버튼을 누를 경우 기울어진 방향의 모터만 RPM이 높아지거나 낮아지게 된다. 이러한 제어로 인해 휠의 RPM차이로 공의 회전이 달라지게 되어 오른쪽으로 휘거나 왼쪽으로 휘어져 발사될 수 있다.

3.4 하단부의 이동 제어

기존의 피칭머신에서 새로 추가 할 하단부의 모터(BL5057), 설계된 제어기가 코딩된 모터 드라이브, 아두이노 그리고 블루투스 모듈을 이용해 제어를 하게 된다. 하단부의 블루투스 모듈과 스마트폰을 테더링 하여 제어를 하게 되는데, 이 때 스마트폰에 내장된 기울기 센서를 이용하여 움직임을 제어 하게 된다. BluCar라는 스마트폰의 어플리케이션을 이용하여 움직임을 제어한다.

BluCar어플리케이션(그림 5(좌))은 스마트폰의 기울기 센서를 이용하여 좌회전 우회전이 가능하도록 설정이 되어있다. Connect to a device를 눌러 블루투스통신을 하고 Forward나

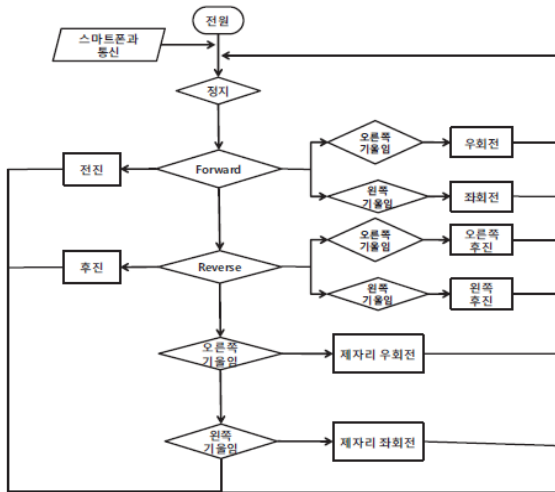


그림 6 하단부 이동 제어 알고리즘
 Fig. 6 Lower part movement control algorithm

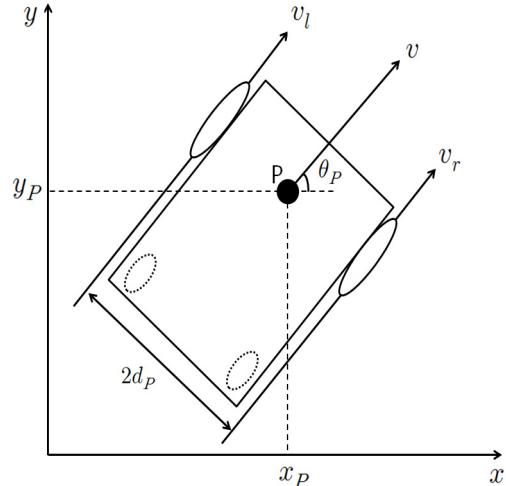


그림 7 하단부 모델
 Fig. 7 Upper body part model

Reverse버튼을 누르거나 버튼을 누르면서 기울이게 되면 코딩된 명령어가 블루투스 무선통신을 통하여 피칭머신에 제어명령을 전달한다. 그러면 피칭머신은 스마트폰으로부터 수신받은 명령패킷을 분석하여 피칭머신의 모터를 구동시킨다. 그림 7에서 하단부의 바퀴가 지면에 대해 미끄러짐이 없다고 가정할 때 수식은 다음과 같다[4].

$$\Delta x_p \sin \theta_p - \Delta y_p \cos \theta_p = 0 \tag{25}$$

여기서 P는 바퀴사이의 중심이고, θ_p 는 하단부의 회전각을 의미한다. 하단부 모델은 논홀로노믹(non-holonomic)이다. 다시 말해 하단부는 한번에 x 또는 y 방향으로 이동할 수 없다. 그 이유는 하단부가 다리가 아닌 바퀴로 되어 있기 때문에 바퀴가 90°로 꺾여 움직이지 않는 이상 한번에 x,y방향으로 이동할 수 없다. 로봇의 선속도 v와 각속도 ω 는 다음과 같다[4].

$$\Delta x_p \sin \theta_p + \Delta y_p \cos \theta_p = v, \Delta \theta_p = \omega \tag{26}$$

식 (26)을 정리하면 다음과 같다.

$$\begin{bmatrix} \Delta x_p \\ \Delta y_p \\ \Delta \theta_p \end{bmatrix} = \begin{bmatrix} \cos \theta_p 0 \\ \sin \theta_p 0 \\ 0 \quad 1 \end{bmatrix} \begin{bmatrix} v \\ \omega \end{bmatrix} \tag{27}$$

따라서 v, ω 는 다음과 같이 구해진다.

$$v = \frac{v_r + v_l}{2}, \omega = \frac{v_r - v_l}{2d_p} \tag{28}$$

여기서 v_l, v_r 은 각각 왼쪽과 오른쪽 모터의 PWM(Pulse Width Model)값이고 $2d_p$ 는 양쪽 바퀴 사이의 수평거리이다. 스마트폰에 의해 제어되는 하단부는 스마트폰으로부터 수신 받은 명령패킷에 의해 하단부의 양쪽 DC모터를 제어하며, BluCar어플리케이션에서 Turn Headlights On과 Forward 사이의 숫자는 기울기 정도를 나타내며, 이 기울기는 스마트폰 내부의 기울기 센서를 이용하여 나타내어 진다. 수평상태가 0이라면 오른쪽으로 기울었을 경우 음수로 왼쪽으로 기울었을 경우에는 양수로 나타 내어 진다. Forward를 누릴 경우 양쪽 모터에 신호를 주어 전진을 하게 되고 Reverse는 후진, 오른쪽으로 기울일 경우 오른쪽 모터에만 신호가 들어가 제자리 우회전을 하고, Forward를 누르면서 오른쪽 또는

왼쪽으로 기울일 경우 양쪽모터에 신호가 전달되지만 모터의 속도차이로 인해 우회전 및 좌회전을 하게 된다. 후진의 경우도 마찬가지로 작동을 하게 된다.

4. 시뮬레이션 및 실험 결과

4.1 시뮬레이션

설계된 ϵ -PID제어기를 시뮬레이션한 것이다. 식 (9)에 대한 파라미터 값은 $B=0.001$, $J=0.01$, $k_f\phi_f=3$, $k_e\phi_f=3$, $R_u=1$, $L_u=1$ 이다. ϵ 값은 0.5, 0.2, 0.1에 대해 그래프로 나타내었다. 그림 8은 ϵ -PID제어기가 PID 이득값 튜닝요소인 ϵ 의 변화에 대한 시뮬레이션 결과이다.

그림 8의 시뮬레이션에는 rpm이 500에 수렴 되게 하여 나타낸 것이다. PID이득값 튜닝요소 ϵ 값에 따라 시뮬레이션 결과가 달라지는 것을 보여 준다. 그림 8은 T_L 이 상수일 때의 그래프 이다. ϵ 이 0.5일 때 수렴은 하지만 오버슈트가 존재해 ϵ 이 0.1 일 때 보다 좋지 않은 결과를 보여준다. 그림 9는 $T_L = e^{-t} + 12\sin t + 3\cos t$ 로 설정했을 때 시뮬레이션 결과이다. 이때의 시뮬레이션은 상수일 때보다 결과가 좋지 않지만 그래도 목표치에 수렴을 하고 있는 것을 보여준다. 시뮬레이션에서 보여주듯이 ϵ 이 0.1일 때 좋은 결과 값을 보여주지만 모터의 성능과 연결된 휠을 같이 돌리기 때문에 과부하가 걸리게 되어 본 실험에서는 ϵ 이 0.2일 때 과부하가 생기지 않아 이번 실험을 통해 최종결정 하였다.

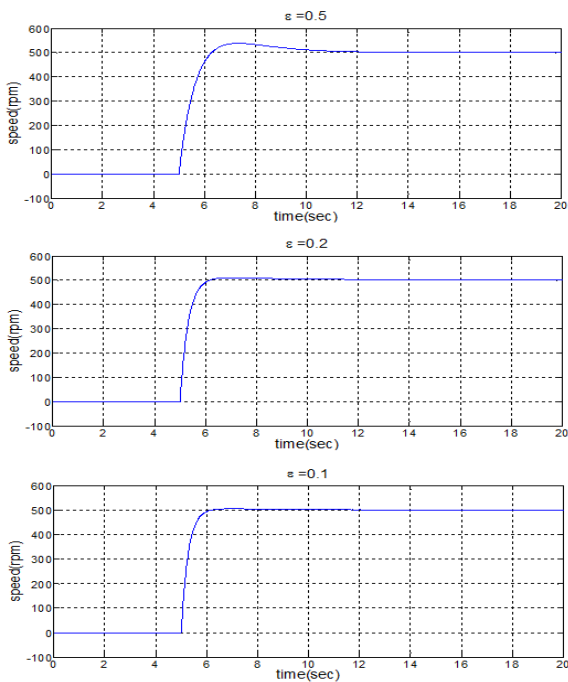


그림 8 T_L 값이 상수인 경우 ϵ 값에 따른 ϵ -PID제어기 시뮬레이션 결과

Fig. 8 T_L is a constant in the simulation result

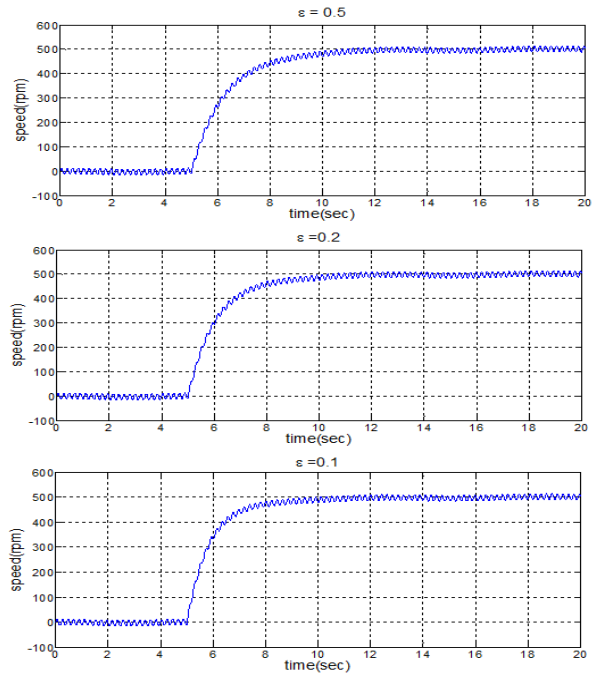


그림 9 T_L 값이 상수가 아닌 경우 ϵ 값에 따른 ϵ -PID 제어기 시뮬레이션 결과

Fig. 9 T_L is time-vary in the simulation result

4.2 실험 결과

본 논문의 피칭머신은 아두이노와 블루투스 모듈, 그리고 안드로이드 기반의 스마트폰을 이용함으로 원격으로 피칭머신의 하단부 이동 제어 및 상단부의 속도 제어를 하게 된다. 하단부의 이동제어는 BluCar 어플을 이용하게 되며, 블루투스 모듈에서는 각각 버튼의 명령패킷을 분석하고 그 데이터를 받고 분석하여 모터드라이브에 설계된 제어기를 통해 각각 모터에 신호를 보내게 되어 상황에 따른 움직임을 보여준다.

그림 10(좌)는 설계된 ϵ -PID에 이득값이 입력되어 컨트롤러를 조절하여 모터가 구동하는 사진이다. 정지와 구동이 되고 있다는 것을 보여주기 위해 왼쪽 모터는 정지된 상태이다. 그림 10(우)는 여러 가지 신호 중 제자리 우회전의 신호를 받은 사진이며 왼쪽 위에서부터 오른쪽으로 진행되며 원격으로 조종하여 하단부의 움직임이 제어가 되고 있다는 것을 보여준다. 그림 11의 그래프는 버튼을 누른 시간에 따른 모터RPM의 변화 이다. 시뮬레이션과는 다르게 살짝 늦게 RPM이 증가하는 것을 보여준다.

그 이유는 모터의 회전축에 샤프트 및 휠이 설치되면서 모터에 부하토크가 생겨 시뮬레이션과 다른 결과를 보여 준다. 추가된 무게 때문에 부하토크가 생겨 갑자기 RPM이 증가하게 되면 모터에 과부하가 걸리게 되어 모터가 정지하거나 사고가 날 수 있기 때문에 컨트롤러에 목표치의 수렴속도를 늘리게 되었다. 그리고 그림 11의 오른쪽 그래프는 모터RPM에 따른 공의 속도이다. 1200이하로 공을 발사했을 때 공의 속도가 거의 측정이 되지 않아 1200부터 측정하여 나타낸 것이다. 그림 9와 그림 11, 12의 실험을 통해 본 논문에서 제안된 스마트폰을 이용하여 피칭머신의 하단부, 상



그림 10 모터 구동(좌), 제자리 우회전(우)
 Fig. 10 Motor operation(left), right turn motion(right)

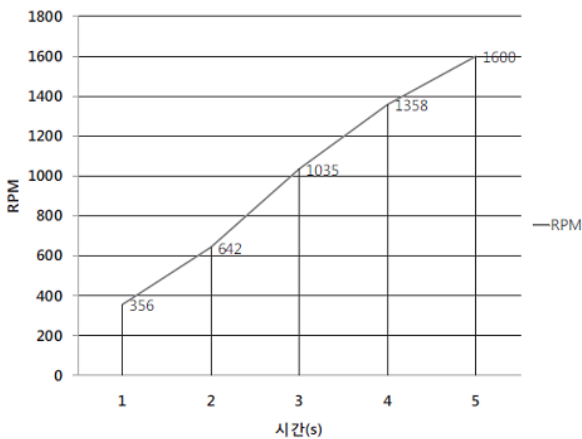


그림 11 시간에 따른 모터 RPM의 변화
 Fig. 11 Change of motor RPM in time

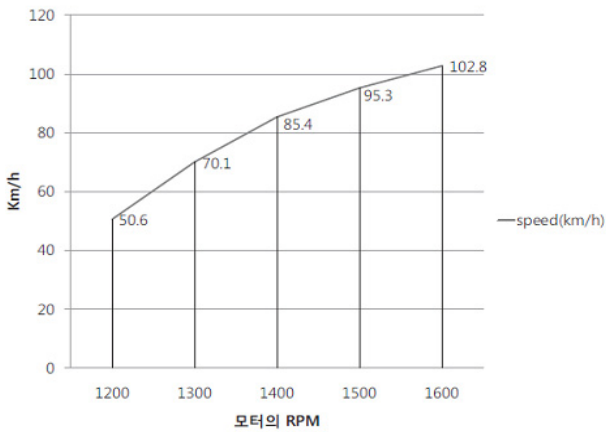


그림 12 모터RPM에 따른 공의 속도변화
 Fig. 12 Velocity of ball correspond to motor RPM

단부 제어가 되고 있음을 알 수 있다.

5. 결 론

본 논문에서는 안드로이드 기반 스마트폰을 사용하여 피칭머신의 이동 및 속도 제어를 구현한다. 피칭머신을 제어하기 위하여 블루투스 모듈과 스마트폰 그리고 제어할 모터

만 있다면 구현을 할 수 있으며, 아두이노로 인하여 복잡한 소스코드대신 간편하게 프로그램을 코딩하여 스마트폰과 블루투스 모듈을 연결 및 모니터링을 할 수 있다. 실험 결과로서 기존 피칭 머신의 특성인 수동적인 부분을 최소화 시켰으며 운동장이나 실내의 야구 연습장등 평평한 바닥을 가지는 공간에서는 어디든지 편하게 움직일 수 있으며, 휠사이의 간격(60mm~110mm)도 줄이거나 넓힐 수 있어 야구뿐만이 아닌 테니스나 구름 구형의 물체를 투척할 수 있다. 그리고 시뮬레이션 및 실제 실험을 통해 ϵ -PID제어기의 유효성을 검증하였고 실험을 통하여 피칭머신의 이동이 가능하며 기존의 특성들을 최소화하여 효율이 증가 하였음을 알 수 있다.

감사의 글

본 논문은 동아대학교 학술연구비 지원에 의하여 연구되었음.

References

- [1] S. H. Kim, "DC, AC, BLDC motor control", bogdoo publishing company, vol. 3, 2012.
- [2] S. S. Gwak, Y. H. Seo, and T. G. Yang, "Remote control of mobile robot base on smart phone environment and its application", 2010 25th ICROS Annual Conference, vol. 39, pp. 431-435.
- [3] J. H. Kwak and J. H. Lim, "Bluetooth network for mobile system control", Journal of Institute of Control, Robotics and Systems (in Korean), vol. 10, no. 11, pp. 1052-1057, 2004.
- [4] Y. G. Kim and D. H. Kim, "Development of a personal riding robot controlled by a smartphone based on android os", Journal of Institute of Control, Robotics and Systems (in Korean), vol. 19, no. 7, pp. 592-598, 2013.
- [5] O. Yang and H.-W. Je, "Implementation of the monitoring system for power condition system(PCS) using a smartphone and bluetooth communication", Journal of the Korea Institute of Information and Communication Engineering, vol. 16, no. 10, pp. 2185-2191, 2012.
- [6] H. An, H. Gao, S. Lu, and J. Tan, "A design of intelligence interactive system of service robot based on th android", IEEE International Conference on Robotics and Biomimetics, pp. 2345-2350, Dec. 2012.
- [7] ARDUINO, <http://www.arduino.cc>
- [8] H.-L. Choi, Y.-H. Chang, Y. H. Oh, and J.-T. Lim, "On robust position control of DC motors by ϵ -PID controller and its application to humanoid robot arms", IFAC, pp. 14449-14453, July. 2008.
- [9] S. V. Delden and A. Whigham, "A bluetooth-based architecture for android communication with an

articulated robot", 2012 International Conference on CTS, pp. 104-108, Denver, CO, May.

- [10] S. W. Moon, Y. J. Kim, H. J. Myeong, C. S. Kim, N. J. Cha, and D. H. Kim, "Implementation of smartphone environment remote control and monitoring system for android operating system-based robot platform", 8th International Conference on URAI, pp. 211-214, Nov. 2011.

저 자 소 개



박성용 (朴成龍)

2013년 동아대학교 전기공학과 졸업
2012년 ~ 현재 동 대학원 석사과정 재학
중 관심분야는 입-출력 궤환선형화 및
로봇 제어 분야 등



최호림 (崔虎林)

1996년, Univ. of Iowa, USA(학사). 1999
년, 2004년 KAIST(석·박사). KAIST 정
보전자연구소 연구조교수, KIST 인지로
봇연구단 선임연구원, 2007년~현재 동아
대학교 전기공학과 부교수, 관심분야는
비선형 시스템의 궤환선형화 기법, 시지
연 시스템, DEFS 등.