

영상 정보를 이용한 ROBOKER 팔 위의 역진자 시스템의 지능 밸런싱 제어 구현

Intelligent Balancing Control of Inverted Pendulum on a ROBOKER Arm Using Visual Information

김정섭¹ · 정 슬^{2*}

Jeong Seop Kim and Seul Jung

1삼성전자

2충남대학교 메카트로닉스공학과

요 약

본 논문에서는 영상 정보를 이용하여 로보커 팔위의 역진자의 밸런싱 제어를 한다. 로봇 팔위에 놓인 역진자의 각도는 카메라로 검출하고 검출된 각도 값은 제어기로 귀환되어 오차를 생성한다. 따라서 전체 제어루프는 폐회로 루프를 형성한다. 제어 성능을 높이기 위해 기존 선형제어기에 신경망 제어기를 더하였다. RBF 네트워크의 학습 알고리즘은 FPGA에 설계된 부동소수점 연산이 가능한 디지털 제어기에 의해 수행된다. 실험을 통하여 전체 시스템 성능을 검증하였다.

키워드 : 역진자, 로봇 팔, 비주얼 서보잉, 신경망 제어

Abstract

This paper presents balancing control of inverted pendulum on the ROBOKER arm using visual information. The angle of the inverted pendulum placed on the robot arm is detected by a stereo camera and the detected angle is used as a feedback and tracking error for the controller. Thus, the overall closed loop forms a visual servoing control task. To improve control performance, neural network is introduced to compensate for uncertainties. The learning algorithm of radial basis function(RBF) network is performed by the digital signal controller which is designed to calculate floating format data and embedded on a field programmable gate array(FPGA) chip. Experimental studies are conducted to confirm the performance of the overall system implementation.

Key Words : Inverted pendulum, robot arm, visual servoing, neural network control

1. 서 론

인간의 눈에 해당하는 영상정보는 로봇에 있어서 반드시 필요한 센서이다. 이동로봇이 장애물을 발견하고 피해가거나 로봇 팔이 물체를 가지고 작업을 하는 경우와 같이 물체를 인식하는데 꼭 필요한 센서이다. 영상 센서는 장애물 감지의 경우에 레이저 센서와 함께 사용되어 정확한 물체의 정보를 제공한다.

이동로봇의 자기 위치 인식 및 장애물 회피의 경우, 레이저나 초음파 센서가 제공하는 라인 정보와 달리 평면이나 입체 정보를 제공하는 영상센서는 매우 효율적이다. 특히,

근거리의 물체나 주변 환경의 전체적인 윤곽을 쉽게 알아내기 위해서는 스테레오 비전을 사용하면 효과적이다. 영상에서 제공하는 3차원 정보는 주변의 지도를 작성하는데 사용되어 SLAM에 사용되고 있다.

로봇 팔의 작업에도 영상 센서가 사용되고 있는데, 작업 공간에 놓인 물체의 좌표와 형태를 인식하여 조작하는 비주얼 서보잉 제어방식을 수행하도록 해준다. 비주얼 서보잉 방식은 로봇이 기능화하는 필수적인 기술로 로봇은 단순한 pick and place의 위치제어 위주의 기능에서 벗어나 협조제어와 같은 고난위도의 작업을 수행하도록 해준다. 특히, 서비스 로봇의 경우 두 팔로 작업하는 것이 필요한데 위치제어를 포함하는 힘제어 방식을 수행하기 위해 비전 정보가 매우 중요하게 대두된다[1,2].

특히, 물체가 움직이는 경우에 로봇이 작업을 하려면 카메라를 통해 물체의 위치를 정확하게 알아내는 것이 중요하다. 실시간 물체의 위치 정보는 로봇 팔이 물체를 따라 움직이도록 하는 위치 명령어가 되고 로봇 팔은 위치를 추종하도록 제어된다. 따라서 비주얼 서보잉 제어를 수행하기 위해서는 물체의 위치를 얻어내는 영상처리, 로봇팔의 위치를 나타내는 기구학, 그리고 로봇의 움직임을 제어하는 제어기에 대한 구현이 필요하다[3].

접수일자 : 2011년 3월 27일

완료일자 : 2011년 8월 25일

* : Corresponding author

감사의 글 : 본 연구는 이 논문은 교육과학기술부 특정기초사업(R01-2008-000-10992-0)과 2010년 충남대학교 학술진흥재단에서 지원하여 연구하였으며 지원에 감사드립니다.

역진자 시스템제어는 비선형 제어의 대표적인 것으로 선형연구를 통해 DSP를 사용하여 신경망 제어기의 성능을 검증하였다 [4,5]. 하지만, 본 논문에서는 플로팅 포인트 형태의 데이터 연산이 가능한 프로세서를 FPGA에 설계하여 신경회로망 제어기를 system-on-chip 개념으로 구현하였다. 신경회로망을 실시간으로 학습하기 위해 FPGA에 부동소수점 연산이 가능한 디지털 프로세서코어를 설계하였다. 이 코어를 사용하여 지능제어 알고리즘과 주변 하드웨어를 처리하도록 디지털 신호처리 제어기를 설계하였다 [6-11].

구현한 제어기의 성능을 검증하기 위해 영상 정보를 이용한 로봇 팔의 움직임을 실시간으로 제어하기 위해 비주얼 서보잉 제어 루프를 구성하였다. 로봇 팔 끝에는 역진자가 달려있어 제어가 되지 않으면 쓰러진다. 로봇의 머리에 달려있는 카메라는 로봇의 역진자 끝에 달려 있는 빨간색 무게 추로부터 뿜어낸 각도를 얻게 된다. 이 무게 추의 영상이 기준 축으로부터 벗어나게 되면 벗어난 만큼의 오차가 발생되어 로봇은 이를 보상하기 위해 움직이게 되는데 이는 제어기가 담당한다. 제어기로는 RBF 신경회로망 제어기를 사용하였다.

실제로 구현한 하드웨어 및 소프트웨어 시스템의 성능을 알아보기 위해 실제 구현된 FPGA 기반의 임베디드 시스템을 로봇의 비주얼 서보잉 역진자 뿜어낸 제어 시스템에 적용하여 실험을 통해 그 성능을 검증하였다.

2. 로봇 기구학

로봇 팔의 위치 제어를 위해서 실험에 사용되는 휴머노이드 로봇 ROBOKER의 기구학적 분석이 필요하다. 로봇 팔의 각 축의 좌표계의 설정은 Denavit-Hartenberg의 표현 방식을 적용하였다.

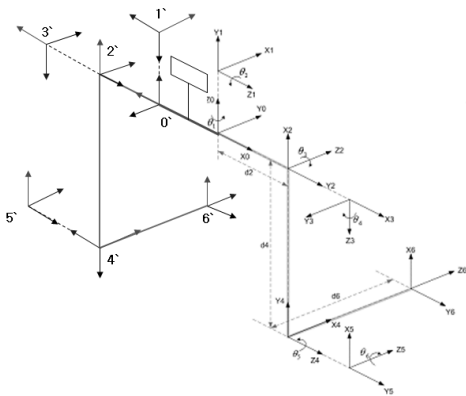


그림 1 로봇 팔의 D-H 변수 및 좌표
Fig. 1 Coordinate representation of D-H parameters

로봇 end-effector의 좌표값과 회전 변환 값을 알 경우, 역기구학 해석 방법에 따라 이 로봇의 역기구학을 구할 수 있다. 그러나 로봇의 좌표값만을 알고 회전 변환 값을 모르는 경우 이 방식을 적용하여 역기구학을 구할 수 없게 된다. 따라서 아래와 같은 구속조건을 줌으로써 이러한 문제점을 해결하고 역기구학을 구할 수 있다[3,12].

1: x_0 축과 링크 d_2 가 이루는 각은 x_0 축과 원점에서

px_6, py_6 가 이루는 직선과의 각의 $\frac{1}{2}$ 이다.

2: 링크 d_4 와 d_6 를 x_0y_0 평면에 투영하면 링크 d_2 의 끝점과 (px_6, py_6) 가 이루는 직선 위에 있다.

3: θ_6 는 로봇 end-effector의 좌표값에 영향을 미치지 못하므로 영으로 놓는다.

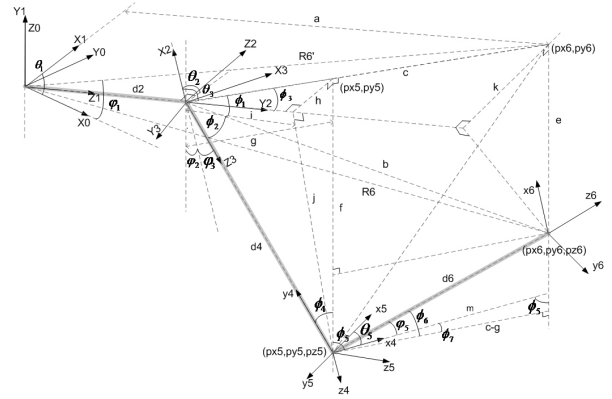


그림 2. 각 조인트의 기하학 분석
Fig. 2 Graphical analysis of each joint

3. 제어방식

1. 선형 제어

일반적인 n축 로봇의 동역학 모델은 다음과 같다.

$$D(q)\ddot{q} + C(q, \dot{q})\dot{q} + G(q) + f_f(\dot{q}) = \tau \quad (1)$$

여기서, q 는 $n \times 1$ 조인트, τ 는 $n \times 1$ 벡터의 토크, $D(q)$ 는 $n \times n$ 관성 행렬, $C(q, \dot{q})\dot{q}$ 는 $n \times 1$ 벡터의 Coriolis와 원심력, $G(q)$ 는 $n \times 1$ 벡터의 중력, f_f 는 $n \times 1$ 벡터의 마찰력을 나타낸다.

로봇의 PD 제어의 경우, 제어 법칙은 다음과 같다.

$$u_f = K_D \dot{e} + K_P e \quad (2)$$

여기서, K_P, K_D 는 제어기 이득값이고 $e = q_d - q$ 이다.

2. 신경회로망 제어

제어 성능을 향상시키기 위해 PD 제어기 기반에 신경회로망을 이용하여 외부에서 보상하는 구조로 제어기를 설계하였다. 일반적인 Feedback error learning 방식의 신경회로망 제어기는 그림 3과 같다[6].

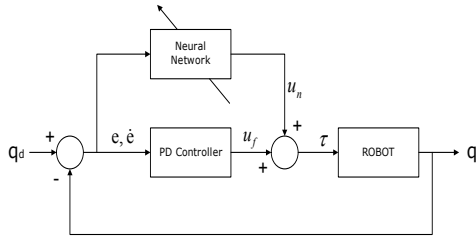


그림 3 신경회로망 제어의 블록디아아 그림
Fig. 3 Block diagram of neural network controller

제어 입력은 PD제어기와 신경망 제어기의 출력을 합한 것으로 다음과 같다.

$$\tau = u_f + u_n \quad (3)$$

여기서 u_n 는 신경회로망으로부터의 제어 보상 신호이다.

사용된 신경회로망은 그림 4의 radial basis function (RBF) 기반의 구조를 가지고 있다. RBF는 구조적으로 간단하며 학습속도가 빠르다. 은닉층의 비선형 함수는 유클리디안 거리에 의해 학습하는 것이 특징이다. 또한 출력층의 구조가 선형이므로 수학적 분석이 용이하다는 장점이 있다.

은닉층에 가우시안 함수를 통과함수로 사용하여 비선형성을 주게 된다.

$$\phi_j(X) = \exp\left(-\frac{\|X - \mu_j\|^2}{2\sigma_j^2}\right) \quad (4)$$

여기서 X 는 입력 벡터이고, μ_j 는 j 번째 은닉층 뉴런의 평균값이며, σ_j 는 j 번째 은닉층 뉴런의 공분산 값이다. K 번째 출력층의 값은 다음과 같이 계산될 수 있다.

$$y_k = \sum_{j=1}^M \phi_j w_{jk} + \theta_k \quad (5)$$

여기서, ϕ_j 는 j 번째 은닉층 뉴런의 출력이며, w_{jk} 는 j 번째 은닉층과 k 번째 출력층 사이의 가중치이다. θ_k 는 k 번째 출력층의 바이어스이다.

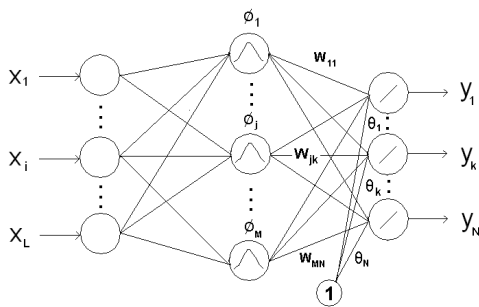


그림 4 RBF 신경회로망 구조
Fig. 4 The RBF neural network structure

출력층의 각 뉴런의 결과는 각각의 기준 값과 비교되어 오차를 생성해 낸다.

$$e_k = yd_k - y_k \quad (6)$$

여기서 yd_k 는 기준 입력 값이다.

역전과 알고리즘은 오차를 최소화하려는 방향으로 가중치값을 업데이트 한다. 은닉층과 출력층 사이의 가중치 값들과 비선형 함수의 변수값들이 매번 새롭게 된다. 역전과 알고리즘을 전개하기 위한 목적함수는 다음과 같다.

$$E = \frac{1}{2} u_f^T u_f \quad (7)$$

여기서 역전과 알고리즘은 다음과 같은 그라디언트 디센트 알고리즘을 사용하여 목적함수를 최소화 시키게 된다.

$$\Delta w = -\eta \frac{\partial E}{\partial w} \quad (8)$$

여기서 w 는 가중치이며, η 는 학습률이다. 위의 식을 통해 각각의 가중치를 업데이트하는 식을 유도하면 다음과 같다.

$$\begin{aligned} \Delta w_{jk} &= \eta_w e_k \phi_j, \\ \Delta \theta_k &= \eta_\theta e_k \\ \Delta \mu_j &= \eta_\mu \phi_j \sum_{k=1}^{N_o} \frac{(x_i - \mu_j)^2}{\sigma_j^3} \sum_{k=1}^{N_o} e_k w_{jk} \\ \Delta \sigma_j &= \eta_\sigma \phi_j \sum_{k=1}^{N_o} \frac{(x_i - \mu_j)}{\sigma_j^2} \sum_{k=1}^{N_o} e_k w_{jk} \end{aligned} \quad (9)$$

여기서 η 는 학습률이다.

4. FPGA 기반 임베디드 설계

1. 부동소수점 연산 모듈 설계

지능형 로봇 제어에 있어서 보다 효율적인 시스템을 설계할 목적으로 FPGA 기반의 원칩 프로세싱 시스템을 설계 및 구현하였다. 연산의 정확성을 위해 32비트 부동소수점 기반의 연산기를 설계하였으며, 파이프라인 구조의 설계를 통해 한정된 FPGA 자원으로 원하는 용량의 제어기를 구현할 수 있도록 하였다. 또한 구현된 프로세서를 동작시키기 위한 전용 명령어 셋을 설계하였으며, 이를 활용하여 프로그래밍이 가능하도록 어셈블러 또한 설계하였다.

그림 5는 설계된 프로세서 내부의 구조를 간단하게 보여주고 있다. 전체적인 구성은 프로그램 메모리, 데이터 메모리, 명령어 레지스터, 명령어 디코더, Instruction Dispatcher, Functional Unit, 레지스터 그룹, Reference ROM, 상태 레지스터, 프로그램 카운터, Address Generator 그리고 몇 개의 멀티플렉서로 이루어져있다. 각각 모듈에 대한 설명은 다음과 같다.

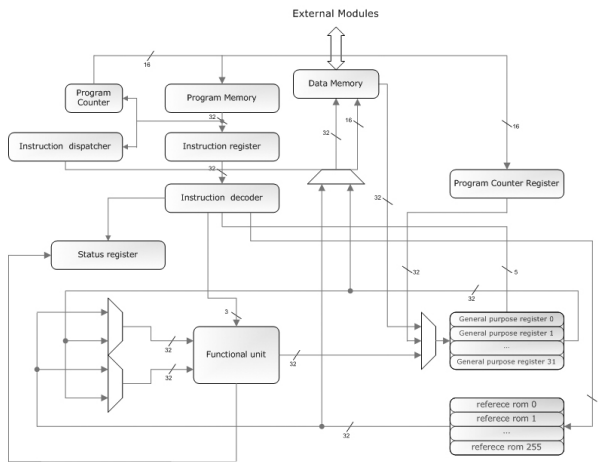


그림 5. 부동소수점 연산기의 블록 다이어그램
Fig. 5. Block diagram of floating-point processor

1) 프로그램 메모리: instruction code를 담고 있는 이 모듈은 ROM 형태로 이루어져있으며 32bit 길이의 word가 총 65,536개로 총 16bit의 address를 가진다. 따라서 Program Memory는 256Kbytes의 용량을 가진다.

2) 데이터 메모리: 연산에 필요한 데이터가 저장되는 RAM 타입의 모듈이다. Program memory와 마찬가지로 1 word가 32bit로 구성되어 있으며 13bit의 address를 가지며 32Kbytes의 용량을 가진다.

3) 프로그램 카운터: 다음에 수행되어야 할 명령어가 들어있는 Program Memory의 주소를 지정한다. Status Register를 참조하여 Jump 명령어이나 Condition Branch 명령어를 수행하기도 한다.

4) 명령어 레지스터: instruction을 instruction decoder에 보내기 전에 1cycle을 지연 시키는 역할을 한다.

5) Instruction Dispatcher: instruction을 Fetch 하는 과정에서 LD 명령어와 같은 일부 명령어의 경우 메모리와 프로세서 사이에서 동작과의 동기화를 위해 Decode 단계 이전에 명령어 해석이 이루어지게 되며 Instruction Dispatcher에서 이러한 역할을 담당한다.

6) Instruction Decoder: 정의되어 있는 32bit instruction code를 opcode와 operand의 주소 등으로 구분하여 해석하고 명령어 수행을 위한 코어 내부의 각 모듈을 제어하기 위한 컨트롤 시그널을 내보낸다.

7) Status Register: arithmetic unit의 연산 결과에 따른 결과나 logical unit의 명령 수행 결과 또는 코어 내부의 상태를 나타내기 위한 시스템 내부 상태 레지스터이다.

8) 레지스터 그룹: Register Group은 32비트 레지스터로 구성된 32개의 레지스터가 존재하며, Multi-Read/ Single-Write 구조를 가진다.

9) Reference ROM: 연산에서 자주 참조되는 상수들, 예를 들어 π 나 자연 상수 e 또는 비선형 함수의 수치해석적 연산을 위해 필요한 계수 값들이 저장되어 있다.

10) Functional Unit: Single precision 기반의 수치 연산 및 논리 연산 그리고 포맷 변환이 이루어지는 모듈이다. 수치 연산으로는 덧셈, 뺄셈, 곱셈 그리고 나눗셈의 사칙연산을 비롯한 절대값과 같은 부가적인 연산이 수행된다. 논리 연산은 operand1과 operand2의 크기를 비교하는 명령어를 수행한다.

11) Address Generator: IEEE 754 부동소수점 형식으로 저장되어 있는 값을 메모리의 어드레스를 위해 unsigned integer 형태로 변환해준다.

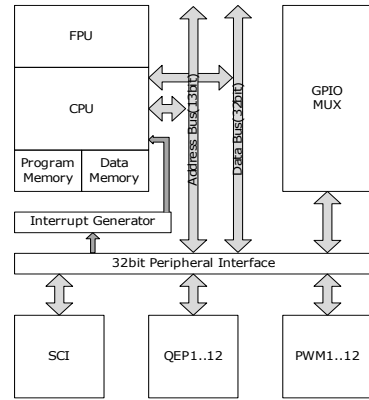


그림 6 DSC의 블록 다이어그램
Fig. 6 Block diagram of DSC

2. DSC 설계

32bit CPU/FPU 기반의 DSC(Digital Signal Controller)를 설계하여 신경회로망 제어기에 필요한 하드웨어 구조를 설계한다. 프로세서 코어를 중심으로 PWM 발생기 그룹 모듈, 엔코더 카운터 그룹 모듈 그리고 SCI 모듈 등이 주변회로로 이루어져 로봇의 모션 제어가 가능하도록 시스템이 설계된다.

그림 6은 설계된 DSC의 구조를 나타내고 있다. 내부는 32비트 버스로 이루어져 있으며, 특정 이벤트가 발생하면 인터럽트가 발생하여 처리하도록 한다.

5. 실험

1. 실험 환경

스테레오 비전 기반 비주얼 서보잉 제어는 영상처리에 대한 내용을 통해 로봇 팔의 좌표를 계산하고, 이를 통해 6축 로봇 팔의 역기구학 식을 통해 각 관절의 제어 기준 값을 구한다. 제어기는 FPGA 기반의 임베디드 시스템에 구현되어 있으며, 스테레오 카메라와 PC 그리고 FPGA의 통합 시스템을 이용하여 휴머노이드 로봇의 팔을 제어한다. 제어의 흐름은 그림 7과 같다.

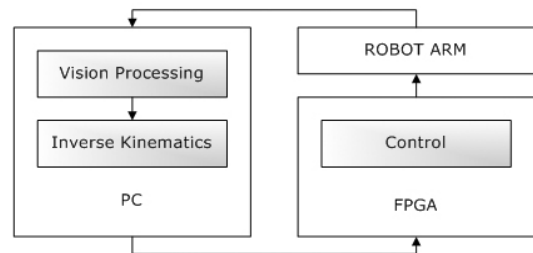


그림 7 비주얼 서보잉 제어의 순서도
Fig. 7 Flow chart of visual servoing control

제어기는 PD 제어기와 신경회로망 기반의 제어기를 사용한다. 로봇 팔 제어에 있어 PD제어기는 안정성을 보장한다. 신경회로망 제어기의 학습률은, $n_w = 0.01$, $n_o = 0.01$, $n_\mu = 0.001$, 그리고 $n_\sigma = 0.001$ 로 설정하였다.

2. 영상처리

비주얼 서보잉을 위한 PC 기반의 영상처리는 OpenCV라는 영상처리 라이브러리를 사용한다. 디지털 이미지에 있어서 낮은 레벨에서 구현되어 있는 Intel IPL(Image Processing Library)와 호환되며, 영상의 양자화, 필터링, 통계적 처리와 OpenCV와 같은 요소들이 하이 레벨 라이브러리로 구현되어 있다.

카메라 제조사인 Point Grey Research에서 제공하는 Digiclops와 Triclops 라이브러리가 지원하는 이미지 Grabbing 함수를 사용하여 카메라로부터 처리 되지 않은 이미지 데이터는 메모리에 저장하게 되며, OpenCV의 함수를 사용하여 특정 조건에 따라 이진화 과정을 거치며, 주변의 잡음을 제거하기 위해 침식 연산을 사용하여 필터링 과정을 거치게 된다. 다시 라벨링 과정을 거쳐 대상의 구분을 해 주게 되면, 이 데이터를 기반으로 면적 중심법으로 좌, 우 카메라에서의 대상의 중심 좌표를 산출하게 되며, Stereo vision에서의 깊이 추출 공식을 사용하여 대상의 Z 축 정보를 산출하게 된다[3].

그림 8은 역진자 무게 추의 위치를 통해 기울어진 거리를 계산하기 위한 기하학 그림이다. 기울어진 거리를 알면 기울어진 각도를 유추해 낼 수 있다.

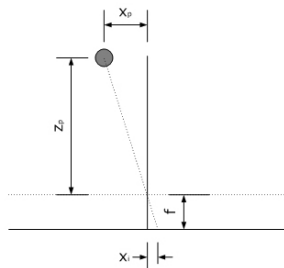


그림 8. 물체의 영상 좌표

Fig. 8 Relation of the object and image plane coordinate

- x_i : image width
- x_p : physical width
- z_p : physical distance
- f : focal length

해상도가 320×240일 때, 사용된 카메라의 CCD의 픽셀당 실제 물리적인 크기는 14.8×14.8μm이다. 만약 거리 f 를 알고 있다면, 카메라 렌즈와 이미지 플랜의 초점 거리인 f 가 6mm이므로, x_p 는 식(10)을 이용해 계산할 수 있다.

$$x_p = \frac{x_i \times z_p \times 14.8 \mu m}{6mm} \tag{10}$$

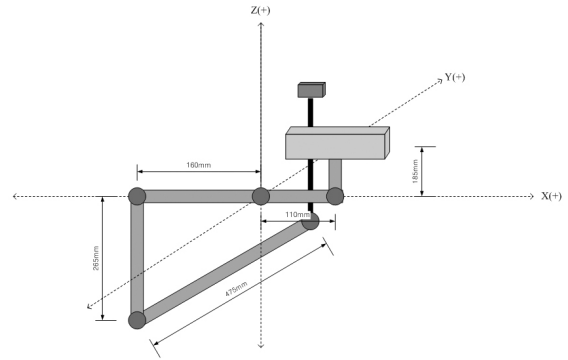


그림 9. 비주얼 서보잉 제어를 위한 좌표
Fig. 9 Setup coordinates for visual servoing control

그림 9는 비주얼 서보잉 제어를 위한 좌표계 설정을 나타내고 있다. 로봇의 왼쪽 팔의 첫 번째 회전축을 글로벌 좌표계의 원점으로 지정하고 기구학 및 역기구학 식을 적용하였다. 따라서, 카메라가 놓여져 있는 x 좌표와 z 좌표만큼의 offset을 적용해야 한다. 실제 제어에서는 트래킹을 하고자 하는 물체의 로컬 좌표계를 영상 처리를 통해 구한 후, 이를 글로벌 좌표계에 적용하게 된다.

3. 밸런싱 실험

그림 10은 비주얼 서보잉 제어를 위한 스테레오 비전 영상 처리를 수행한 결과이다. OpenCV 라이브러리를 사용하여 320x240 해상도의 30Hz의 처리속도를 얻을 수 있었다. 색상 기반의 영상 처리를 통해 역진자의 상단의 빨간색 부분의 위치를 측정한 결과이다.

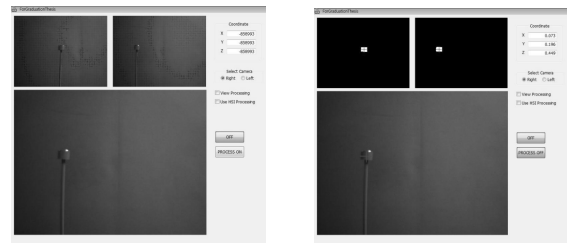


그림 10. 영상 처리 전과 후의 이미지
Fig. 10 Before and after image processing

그림 11에서 보면 역진자는 좌우로 자유롭게 쓰러질 수 있도록 로봇 팔 끝에 달려있다. 만약 손으로 추를 강하게 밀어 무게 추가 영상에서 벗어나면 제어가 되지 않는다.

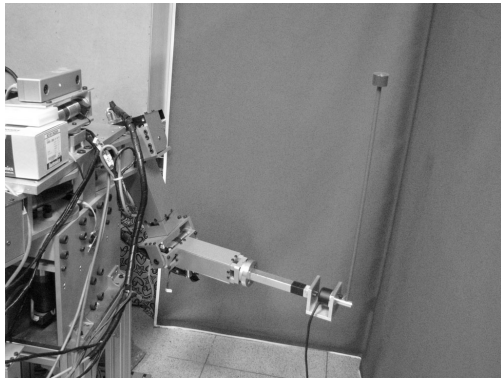


그림 11. 로봇 팔 위의 역진자 제어 실험 환경

Fig. 11 Control of inverted pendulum based on robot arm

그림 11은 FPGA 기반의 신경회로망 제어기와 스테레오 비전 영상처리를 통한 비주얼 서보잉 제어가 이루어지고 있는 그림이다. 신경망 제어기의 경우 오프라인 학습이 없이 온라인으로 학습과 제어가 동시에 이루어진다. 그림 11의 실험은 역진자를 안정화 시키는 것을 보여주고 있다. 그에 상응하는 그래프는 그림 12에 나타나 있다.

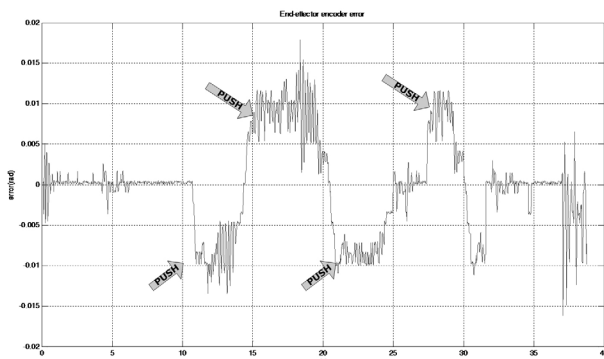


그림 12 역진자 제어 실험결과

Fig. 12 Inverted pendulum control based on robot arm

그림 12는 로봇 팔의 끝 지점에 장착된 엔코더를 통해 얻어진 제어 결과이다. 제어 초기에는 오차가 거의 없지만, 역진자를 일부러 손으로 밀었을 때 (그림 12의 화살표 부분), 오차가 커지면서 최대 0.018rad의 오차가 발생하였지만, 곧 제어가 이루어짐을 알 수가 있다.

6. 결론

본 논문에서는 영상정보를 이용하여 로봇팔위의 역진자 시스템의 균형 제어를 수행하였다. 영상으로 역진자의 각도를 측정하고 신경회로망 제어기를 설계 및 제어하였다. RBF 신경회로망 제어기는 FPGA 기반의 부동소수점 프로세서의 어셈블러 프로그래밍을 통해 설계되었고, 필터 설계나 영상 전처리 동작 등의 수식 연산을 코드화 하였다. 로봇 제어기와 비전 시스템을 상용하여 비전 기반의 비주얼 서보잉 제어 시스템을 구성하고 실험을 통해 그 안정화 성

능을 확인하였다. 외란에도 역진자가 균형을 잘 유지하는 것을 확인하였다.

하지만 이미지의 해상도 및 프레임 속도가 비주얼 서보잉 시스템의 전체 성능에 많은 영향을 미치게 되는 것을 확인하였으며, 이는 앞으로 해결할 숙제로 남는다.

참 고 문 헌

- [1] S. Hutchinson, G. D. Hager, and P. I. Corke, "A Tutorial on Visual Servo Control," *IEEE Trans. on Robotics and Automations*, Vol. 12, No. 5, pp. 651-670, 1996.
- [2] F. Chaumette and S. Hutchinson, "Visual Servo Control Part I: Basic Approaches," *IEEE Robotics and Automations Magazine*, pp. 82-90, 2006.
- [3] 이운규, 김동민, 최호진, 김정섭, 정슬, "스테레오 비전정보를 사용한 휴머노이드 로봇 팔 ROBOKER의 동적 물체 추종제어 및 구현", *제어 로봇 시스템 공학 논문지*, vol.14, no. 10, pp. 998-1004, 2008.
- [4] S. Jung and S. S. Kim, "Hardware implementation of real-time neural network controller with a DSP and an FPGA for nonlinear system", *IEEE Trans. on Industrial Electronics*, vol. 54 no. 1, pp. 265-271, 2007.
- [5] S. Jung, H. T. Cho, and T. C. Hsia, "Neural network control for position tracking of a two-axis inverted pendulum system: Experimental studies", *IEEE Trans. on Neural Network*, vol. 18, no. 4, pp. 1042-1048, 2008.
- [6] M. Miyamoto, M. Kawato, T. Setoyama, and R. Suzuki, "Feedback error learning", *Neural Networks*, vol.1, pp. 251-265, 1988.
- [7] S. Himavathi, D. Anitha and A. Muthuramalingam, "Feed-forward neural network implementation in FPGA using layer multiplexing for effective resource utilization", *IEEE Trans. on Neural Networks*. Vol. 18, no. 3, pp. 880-888, 2007.
- [8] Y. Taright and M. Hubin, "FPGA implementation of a multilayer perceptron neural network using VHDL", *Proc. of ICSP*, pp. 1311-1314, 1998.
- [9] J. Zhu, G. J. Milne, and B. K. Gunther, "Towards an FPGA Based Reconfigurable computing environment for neural network implementations", *IEE Proceedings of Artificial neural networks*, pp. 661-666, 1999.
- [10] I. Mezei and V. Malbasa, "Formal specification of an FPGA based educational microprocessor", *Proc. of International Conf. on Microelectronics*, pp. 667-670, 2002.
- [11] M. Krips and T. Lammert, and A. Kummert, "FPGA implementation of a neural network for a real time hand tracking system", *IEEE*

International workshop on electronic design, test, and applications, 2002.

[12] D. H. Song, W. K. Lee and S. Jung, 'Kinematic analysis and implementation of a motion-following task for a humanoid slave robot controlled by an exoskeleton master robot', *IJCAS*, vol.5, no. 6, pp.681-690, 2007.

저 자 소 개



김정섭 (Jeong-Seob Kim)

2006년 : 충남대 메카트로닉스공학과 졸업.

2009년 : 충남대 메카트로닉스공학과 지능로봇 전공 석사졸업

2009년 ~ 2011년 : 도담 시스템스

2011년 ~ 현재 : 삼성전자

관심분야: 지능 시스템의 SoC 구현, 영상처리 및 인식 시스템.

Phone : 042-821-7232

Fax : 042-823-4919

E-mail : insideasuram@gmail.com



정 슬 (Seul Jung)

1988년 : 미국 웨인 주립대 전기 컴퓨터 공학과 졸업.

1991년 : 미국 캘리포니아대 데이비스 전기 컴퓨터공학과 석사 졸업.

1996년 : 미국 캘리포니아대 데이비스 전기 컴퓨터공학과 박사 졸업.

1997년~현재 : 충남대학교 메카트로닉스 공학과 교수

관심분야 : 지능 시스템 및 지능 로봇, 밸런싱 시스템응용, 미래형 지능형 운송시스템, 로봇교육

Phone : 042-821-6876

Fax : 042-823-4919

E-mail : jungs@cnu.ac.kr