



저작자표시-비영리-변경금지 2.0 대한민국

이용자는 아래의 조건을 따르는 경우에 한하여 자유롭게

- 이 저작물을 복제, 배포, 전송, 전시, 공연 및 방송할 수 있습니다.

다음과 같은 조건을 따라야 합니다:



저작자표시. 귀하는 원저작자를 표시하여야 합니다.



비영리. 귀하는 이 저작물을 영리 목적으로 이용할 수 없습니다.



변경금지. 귀하는 이 저작물을 개작, 변형 또는 가공할 수 없습니다.

- 귀하는, 이 저작물의 재이용이나 배포의 경우, 이 저작물에 적용된 이용허락조건을 명확하게 나타내어야 합니다.
- 저작권자로부터 별도의 허가를 받으면 이러한 조건들은 적용되지 않습니다.

저작권법에 따른 이용자의 권리는 위의 내용에 의하여 영향을 받지 않습니다.

이것은 [이용허락규약\(Legal Code\)](#)을 이해하기 쉽게 요약한 것입니다.

[Disclaimer](#)

공학박사 학위논문

안내 로봇 제어를 위한  
새로운 위치 측정 시스템

A New Positioning System  
for Guide Robot Control

2013년 2월

서울대학교 대학원

전기 컴퓨터 공학부

조금배

# 안내 로봇 제어를 위한 새로운 위치 측정 시스템

A New Positioning System  
for Guide Robot Control

지도 교수 이 범 희

이 논문을 공학박사 학위논문으로 제출함  
2012년 10월

서울대학교 대학원  
전기 컴퓨터 공학부  
조 금 배

조금배의 공학박사 학위논문을 인준함  
2012년 12월

위 원 장           최 종 호           (인)

부위원장           이 범 희           (인)

위    원           최 진 영           (인)

위    원           오 성 회           (인)

위    원           박 재 병           (인)

# 초록

로봇에 적용되는 상대적 또는 절대적 위치 인식 시스템은 초음파 센서처럼 음파를 이용하거나 비전(vision), 적외선, 레이저(LASER), 라이다(LIDAR) 센서처럼 빛을 이용하거나 GPS(Global Positioning System), 레이다(RADAR)와 같이 전파를 이용한다. 실내 환경에서 많이 사용되는 초음파, 비전 센서들은 외란(disturbance)의 변화에 민감하여 바람이 거세거나 연무에 의해서 그 성능이 심각하게 저하된다. 본 논문은 이러한 외란의 영향을 중요시해야 하는 로봇, 예를 들어 사람 가까이서 이동하는 안내 로봇이나 복잡한 실내 환경에서 임무를 수행하는 가사 도우미 로봇들이 이용할 수 있는 새로운 위치 측정 시스템의 제안과 개발, 그리고 실험 및 분석에 관한 것이다.

본 논문에서 제안하는 새로운 위치 측정 시스템은 관절각 또는 링크각을 측정할 수 있는 시리얼 링크의 한 끝에 IMU 를 부착한 형태를 가지며 안내 로봇에 적용하여 조이스틱 역할을 하는 것을 “Intelligent Lead”라고 정의한다.

시각장애인이나 노인을 위한 안내 로봇은 스스로 장애물을 회피하면서 사용자의 위치 및 속도 제어 명령을 추종해야만 한다. 또한 사용자의 위치를 파악하고 사용자 근처에서 적당한 거리를 유지함으로써 사용자의 보행에 방해가 되지 않도록 설계 및

제작되어야 한다. 위의 기능 구현을 위해서는 사용자의 제어 명령을 획득하는 기능뿐만 아니라 외란의 영향을 받지 않으면서 사용자의 위치를 실시간으로 파악할 수 있는 센서가 필요해진다. 그러나 현재까지 주로 사용되고 있는 초음파 센서, 비전 센서 및 라이다 센서 등으로는 외란의 영향으로부터 사용자의 안전을 보장하기 어렵다. 이러한 문제를 해결하기 위해서 본 논문에서는 외란의 영향을 받지 않고 가까운 장애물에 대한 거리 정보를 획득할 수 있는 새로운 위치 측정 수단으로써 Intelligent Lead 를 제안한다.

본 논문에서 제안하는 Intelligent Lead 는 로봇과 사용자와의 거리와 속도 제어 량, 회전 제어 량을 실시간 출력하는 새로운 HRI(Human Robot Interaction)용 센서로서 시리얼 링크의 한쪽 끝에 IMU(Inertial Measurement Unit)가 부착되고 시리얼 링크의 각각의 관절(joint)마다 엔코더(encoder)가 부착된 형태를 가지며 로봇에 부착된 LIDAR 로부터 획득되는 사용자의 관절 정보를 융합한다. 시리얼 링크는 사용자 손목의 위치 및 자세를 측정하며 LIDAR 는 사용자의 어깨 및 몸체의 위치를 측정하고 두 센서 데이터를 IMU 와 함께 EKF(Extended Kalman Filter)로 융합(fusion) 함으로써 LIDAR 의 시야각(field of view)을 벗어나는 가까운 거리에서도, 거친 지면으로 인한 로봇의 흔들림에도 안정적으로 사용자의 어깨, 몸체, 손목의 위치를 획득한다. 또한 어깨 및 몸체의 위치를 기준으로 손목의 위치가 정상적인 보행 시의

위치에서 이탈되는 정도를 계산하고 임계 값 이상이 되면 이탈 량에 비례해서 속도와 방향 전환을 위한 제어 량으로 출력한다.

제작된 Intelligent Lead 를 실험을 위해서 제작된 이동 로봇(mobile robot)에 장착하고, 1)Intelligent Lead 를 사용하지 않는 상태에서 로봇이 인공 장애물을 회피하며 목적 지점까지 도달하는 실험과, 2)Intelligent Lead 를 모바일 조이스틱으로 사용하여 사용자가 Intelligent Lead 를 이용해서 로봇을 목적지에 도달시키는 실험과, 3)사용자와 로봇이 협업하여 장애물 회피와 목적지 도달 및 서로간의 간격을 유지하는 실험과, 4)사용자가 Intelligent Lead 만을 잡고 로봇을 따라가며 주어진 목적지에 도달하는 안내견 모사 실험들을 수행하며 위의 4 가지 실험 결과의 분석을 통해서 Intelligent Lead 가 안내 로봇 제어를 위한 위치 측정 시스템으로 사용하기에 적합함을 보여준다.

주요어 : 위치 측정 시스템, 안내 로봇, 시리얼링크, 라이다, 센서융합

학 번 : 98420-868

# 목차

초록 .....	i
목차 .....	iv
표 목차 .....	vi
그림 목차 .....	vii
기호의 정의 .....	ix
제 1 장 서 론 .....	1
1.1 연구의 배경 .....	1
1.2 문제 정의 및 해결 방안 .....	5
1.3 논문의 구성 .....	10
제 2 장 시리얼 링크를 이용한 위치 측정 .....	12
2.1 좌표계와 회전 행렬 정의 .....	13
2.2 쿼터니언에 기반한 AHRS .....	18
2.3 관절각 측정 방식의 Intelligent Lead .....	21
2.4 링크각 측정 방식의 Intelligent Lead .....	26
제 3 장 기존의 위치 측정 시스템들과의 데이터 융합 .....	32
3.1 6-DOF 강체 운동 방정식 .....	33
3.2 GPS/INS Integration .....	37
3.3 시리얼 링크/INS Integration .....	45
3.4 Intelligent Lead의 성능 지수 .....	49

제 4 장 Intelligent Lead를 이용한 지능형 안내 시스템.....	52
4.1 Intelligent Lead 출력값 생성 알고리즘 .....	53
4.2 이동 로봇의 모션 제어 알고리즘 .....	59
4.3 실험 시스템 구성 .....	64
제 5 장 실험 결과.....	70
5.1 센서 출력값 측정 및 분석 .....	72
5.2 장애물 회피 실험 및 분석 .....	74
5.3 모바일 조이스틱 실험 및 분석.....	77
5.4 사용자와 로봇의 협업 실험 및 분석 .....	80
5.5 안내건 모사 실험 및 분석 .....	82
5.6 실험 결과 비교 정리.....	87
제 6 장 결론 및 향후 연구 계획 .....	90
6.1 결론 .....	90
6.2 논문의 기여도.....	91
6.3 향후 연구 방향 .....	91
참고문헌.....	93
Abstract .....	98



## 표 목차

표 2.1	롤/피치/요우 회전 축과 기본 회전행렬 정의 .....	13
표 2.2	대표적인 DCM들.....	14
표 2.3	DCM의 수식적 관계 .....	16
표 2.4	시리얼 링크의 D-H 인자들.....	23
표 3.1	GPS/INS Integration 관련 기호 정의 .....	38
표 4.1	Intelligent Lead 출력 관련 기호 정의.....	53

# 그림 목차

그림 1.1 Intelligent Lead의 사용 예.....	8
그림 2.1 동차 변환 행렬의 비교 .....	17
그림 2.2 쿼터니언 회전 연산자 .....	20
그림 2.3 쿼터니언과 오일러 각의 상호 변환 관계식 .....	21
그림 2.4 시리얼 링크와 사용된 절대각 엔코더 .....	22
그림 2.5 사용된 IMU 센서와 키넥트 센서.....	25
그림 2.6 링크각 측정 방식의 Intelligent Lead.....	27
그림 2.7 EKF로 구현된 AHRS 블록다이아그램 .....	28
그림 3.1 ECEF 좌표계와 LLH 좌표계 .....	37
그림 3.2 LLH 좌표계와 NED 좌표계 .....	38
그림 3.3 INS 블록다이아그램 .....	39
그림 4.1 사용자 제어 모션 .....	54
그림 4.2 좌표계, 각 그리고 거리 정의.....	55
그림 4.3 법선 벡터 정의.....	56
그림 4.4 Intelligent Lead 블록다이아그램 .....	58
그림 4.5 로봇 모션 제어 알고리즘 블록다이아그램 .....	64
그림 4.6 Hokuyo UTM-30LX 스펙 .....	65
그림 4.7 스타게이저와 랜드마크 .....	66
그림 4.8 안내 로봇 블록다이아그램 .....	67
그림 4.9 이동 로봇에 부착된 Intelligent Lead(앞면) .....	68
그림 4.10 이동 로봇에 부착된 Intelligent Lead(뒷면) .....	68

그림 4.11 모니터링 프로그램 .....	69
그림 5.1 핸들과 사용자의 관절 위치.....	72
그림 5.2 Intelligent Lead 내부 연산값 .....	73
그림 5.3 Intelligent Lead 출력값.....	74
그림 5.4 로봇 트랙 - 장애물 회피 실험 1회.....	75
그림 5.5 로봇 트랙 - 장애물 회피 실험 2회.....	76
그림 5.6 로봇 트랙 - 모바일 조이스틱 실험 .....	78
그림 5.7 Intelligent Lead 출력값 - 모바일 조이스틱 실험 .....	79
그림 5.8 로봇 트랙 - 사용자와 로봇의 협업 실험.....	81
그림 5.9 Intelligent Lead 출력값 - 협업 실험 .....	81
그림 5.10 경유점 및 장애물의 위치.....	83
그림 5.11 로봇 트랙 - 안내견 모사 실험 .....	85
그림 5.12 Intelligent Lead 출력값 - 안내견 모사 실험 .....	86
그림 5.13 입력 변수 적용 표와 로봇 패스 비교.....	89

# 기호의 정의

Symbol	Meaning	Symbol	Meaning
제 2장			
$Q$	쿼터니언	$q_0, q_1, q_2, q_3$	쿼터니언 4원소
$p, q, r$	동체좌표계에서 측정한 회전 속도	$P_x, P_y, P_z$	시리얼 링크의 위치 좌표
$C$	회전변환행렬	$\phi, \theta, \psi$	롤, 피치, 요우 오일러각
$T$	동차변환행렬	첨자 의미 $n, e, b, i$	항법좌표계, 지구 중심좌표계, 동체 좌표계, 관성좌표계

제 3장			
$p = [\phi \quad \lambda \quad h]^T$	위치 벡터, [위도, 경도, 고도]	$\omega_{ie}$	관성좌표계 기준 지구중심좌표계의 회전속도
$v = [v_N \quad v_E \quad v_D]^T$	NED좌표계에서의 속도	$\omega_{en}$	지구중심좌표계 기준 항법좌표계의 회전속도
$\rho = [\phi_e \quad \theta_e \quad \psi_e]^T$	Euler각, [롤, 피치, 요우]	$\omega_{in}$	관성좌표계 기준 항법좌표계의 회전 속도
$f_N, f_E, f_D$	NED 좌표계에서의 힘	$\gamma(\phi, h), \gamma(\phi)$	지역 중력 모델
$R_\phi$	Meridian Radius	$\zeta_g, \eta_g, \delta_g$	공기 마찰력
$R_h$	Transverse Radius	$H$	출력 행렬
$F$	시스템 행렬		

제 4장			
$p_H$	손의 위치	$O_S$	속도 제어 량
$p_{SL}$	왼쪽 어깨의 위치	$O_T$	회전 제어 량
$p_{SR}$	오른쪽 어깨의 위치	$O_D$	로봇과 사용자와의 거리
$p_T$	몸체 위치	$D_{SH}$	손목과 어깨와의 거리
$\theta_E$	팔꿈치 각	$\vec{n}$	법선 벡터
$\theta_V$	수직 각	$L_{UA}$	윗 팔 길이
$\theta_H$	수평 각	$L_{LA}$	아래 팔 길이

# 제 1 장 서 론

## 1.1 연구의 배경

전세계적으로 현재 약 2 억 8 천 5 백만 명이 시각장애를 가지고 있으며 이 중에서 3 천 9 백만 명은 전혀 볼 수 없다고 알려져 있다. 또한 시각장애인의 90%가 개발 도상국에 살고 있다.[19] 한국에서의 1,2 급 시각 장애인의 수는 4 만 5 천명에 달하지만 안내견은 60~70 마리에 불과하다. 또한 고령화로 인한 노인성 치매 및 거동이 불편한 환자의 수도 갈수록 늘어나는 추세이다. 따라서 시각장애인이거나 노인을 위한 안내 로봇의 개발이 절실히 요구되고 있다.

시각장애인이 이동 할 때에 위치나 장애물에 대한 정보를 제공하는 수단으로 지금까지 다양한 방법들이 연구되어 왔다. RFID(Radio Frequency Identification)를 이용한 연구[1,2]로서 RobotCart 는 시각 장애인이 물건을 구매할 때 RFID 로 받아들인 위치정보를 음성으로 바꾸어 전달해준다. 또한 의복에 환경 인식을 위한 각종 센서를 장착하고 환경 정보를 다른 감각기관을 통해서 시각장애인에게 전달하는 항법수단(Navigation tool)에 관한 연구도 진행되어졌다.[3] 시각장애인을 보조하는 이동 로봇(mobile robot)으로 개발된 GuideCane[4], RoJi[5], ROVI[6] 등은 two-wheel-type 이동 로봇에 초음파센서를 탑재하고 이동 로봇이

장애물 회피를 위한 조종행위(steering action)에 의해서 형성된 회전력을 사용자가 핸들을 통해서 피드백(feedback) 받는 구조로 되어 있다. 하지만 로봇과 사용자가 지팡이(cane)이라는 강체로 묶여 있음으로 인해서 지면 충격이 그대로 사용자에게 전해지게 된다. 이와는 대조적으로, eyeDog[7]는 비전(vision) 센서와 LRF(Laser Range Finder)를 이용해서 장애물 회피(obstacle avoidance)기능을 구현했지만 단순히 개 줄과 같은 끈을 잡고 로봇을 따라가는 구조로 되어 있다. 끈의 유연성으로 인해서 로봇이 받는 충격이 사용자에게 전달되지는 않지만 사용자와 로봇 간의 위치 정보를 제공하지는 못한다.

본 논문은 궁극적으로 안내 견을 로봇 화하여 인간 친화형 안내 로봇(human-friendly guide robot)을 제작하는 것을 목표로 한다. 시각장애인 중 전혀 볼 수 없는 사람(blind)은 겨우 15% 이내인 것을 고려하면 단순한 환경 인식 센서 또는 제어 명령 추종자로서의 안내 로봇 개발이 아니라, 장애물 회피나 주행방향 설정에서 사용자와 서로 정보를 주고 받는 협력적인 관계를 가지는 로봇 개발이 중요시 된다. 위의 기능을 가지는 인간 친화형 안내 로봇은 스스로 장애물을 인지 회피하면서 동시에 사용자의 위치를 파악하고, 사용자 근처에서 적당한 거리를 유지함으로써 사용자의 보행에 방해가 되지 않으면서도 사용자의 방향 및 속도 제어 명령을 인지 추종해야만 한다. 위의 기능 구현을 위해서는 사용자의

제어명령을 획득하는 기능뿐만 아니라 사용자의 위치를 실시간으로 파악하는 기능이 필요해 진다.

로봇과 사용자가 강체(rigid body)로 연결되거나 끈으로 연결되는 기존 구조에서 충격이 전달되지 않으면서도 로봇이 사용자의 위치 정보를 획득할 수 있도록 하기 위해서 각각의 관절에 엔코더(encoder)가 부착된 시리얼 링크로 로봇과 사용자를 연결하는 것을 본 논문에서 처음으로 제안한다. 실시간으로 관절각을 구한 후에 순방향 기구학을 풀어서 시리얼 링크 끝 단의 위치 및 자세를 측정한다. 이렇게 구해진 사용자 손목의 위치 및 자세 정보는 사용자 몸체의 위치정보와 함께 방향 및 속도 명령을 인지하는 데에 사용한다.

로봇과 사용자가 함께 이동 할 때 사용자의 팔이 앞뒤로 흔들림으로 인해서 손목의 위치만으로 로봇과 사용자와의 거리를 파악하는 것은 현실적으로 어렵다. 따라서 사용자의 몸체의 위치를 파악할 수 있는 새로운 센서가 필요하게 되며 로봇이 장애물을 인식하기 위해서 사용되는 센서들 중에서 선택적으로 사용 될 수 있다.

로봇의 장애물의 위치 인식 수단으로써 초기에는 초음파 센서를 주로 사용했으나 LRF(Laser Range Finder), LIDAR(Light Detection and Ranging), 비전 등을 이용해서 장애물 인식 및 회피를 위한 제어시스템에 관한 연구들이 다양하게 수행되어 왔다.

최근 마이크로소프트 사의 콘솔 게임기 X-box 로 인해 대중에게 친숙해진 키넥트는 뛰어난 가격 경쟁력으로 인해 로봇 분야에서 3 차원 물체나 환경 측정 용도로 기존에 사용하던 3D LRF 나 고가의 LIDAR 를 대신해서 사용되고 있다. 상용 LIDAR 인 SR-3000 에 비해서 정밀도는 떨어지지만 가격이 현저히 낮기 때문에 본 논문에서도 사용자와 로봇간의 위치를 파악하는 수단으로 키넥트 센서를 사용한다. 그러나 LIDAR 는 단일 센서로서 그 활용성이 크지만 연기나 안개에 의한 급작스런 주변환경의 변화로 가려짐(occlusion)이 발생하거나 거친 지면에 의해서 로봇 몸체가 흔들리는 경우에는 심각한 성능저하를 초래한다. LIDAR 뿐만 아니라 시리얼 링크(serial link)도 엔코더의 소형화로 인해서 정밀도가 떨어지며 링크의 크기가 길어질수록, 관절이 늘어날수록 정밀도가 떨어진다. 따라서 각각의 센서로부터 획득되는 위치정보를 융합하는 것이 필요해진다.

단일 센서를 이용하는 위치 측정뿐만 아니라 여러 센서들의 단점들을 보완하여 위치를 측정하는 방법들이 다양하게 연구되어 왔으며 그 중에서도 GPS/INS Integration 은 실외용 위치 측정 시스템으로 널리 사용되고 있다. GPS 의 원형 공산 오차(CEP: circular error probable )가 수 미터에 이르며 터널이나 나무들에 의해서 하늘이 가려지면 위성 데이터가 끊기는 경우가 발생하므로 이것에 대비해서 INS 시스템과 센서 융합을 함으로써 성능을



향상시킨다. 실내에서는 Visual SLAM 기반의 위치 측정 (localization) 및 지도 제작(mapping)이 이루어진다.

## 1.2 문제 정의 및 해결 방법

상기한 바와 같이 시각장애인이나 노인을 위한 안내 로봇은 스스로 장애물을 회피하면서 사용자의 위치 및 속도 제어 명령을 추종해야만 한다. 또한 사용자의 위치를 파악하고 사용자 근처에서 적당한 거리를 유지함으로써 사용자의 보행에 방해가 되지 않도록 설계 및 제작되어야 한다. 이러한 상황을 고려해서 본 논문 서술의 기초가 되는 현안 문제를 다음과 같이 정의한다.

### 1.2.1 문제 정의

먼저 개발 목표는 다음과 같다.

**[개발 목표]** 로봇과 사용자가 근거리에서 공동 작업이 가능한 안내 로봇 제작을 목표로 한다.

개발 목표 상에서 명시된 공동 작업은 다음과 같이 정의한다.

1. 로봇과 사용자 간의 안전거리를 유지한다.
2. 실시간 장애물을 인지 및 장애물과의 충돌을 회피한다.
3. 정해진 목표점으로 이동한다.

상기 공동 작업을 수행하기 위해서 안내 로봇이 가져야 할 기본 기능은 다음과 같다.

1. 사용자가 원할 때 로봇의 방향과 속도를 변경 시킬 수 있어야 한다.
2. 미리 정해놓은 사용자와 로봇간의 거리를 사용자가 임계 속도 이내로 보행하는 경우에 스스로 유지시킬 수 있어야 한다.
3. 스스로 장애물을 탐지 및 회피할 수 있어야 한다.

위의 기능 구현을 위해서는 사용자의 제어명령을 획득하는 기능뿐만 아니라 외란의 영향을 받지 않으면서 사용자의 위치를 실시간으로 파악할 수 있는 센서가 필요해진다. 그러나 현재까지 주로 사용되고 있는 초음파 센서, 비전 센서 및 라이다 센서 등으로는 외란의 영향으로부터 사용자의 안전을 보장하기 어렵다.

기존 센서들의 한계를 정리하면 다음과 같다.

1. 초음파 센서는 외란의 변화에 민감하여 바람이 거세거나 연무에 의해서 그 성능이 심각하게 저하된다. 3D 데이터를 뽑아내는 데에 있어서 기술적인 장벽이 높다.
2. 비전기반 센서는 연무에 의해서 성능이 저하될 뿐만 아니라 시야각(FOV) 한계로 인해서 가까운거리에서 사용자의 몸체와 손목의 위치를 동시에 측정하는데 한계가 있다.
3. 3D LRF 는 고가장비일뿐만 아니라 위상측정(phase-detect) 방식은 실시간 샘플링(real-time sampling)이 불가능하며 TOF(Time-Of-Flight) 방식의 경우에는 빛의

속도제한으로 인해서 몇 cm 이내의 에러를 가지도록 개발하는 것이 현실적으로 어렵다.

4. 전파기반 센서는 차량용 후방감지 레벨 측정용으로 나온 예를 제외하고 수 미터 이내에서 거리측정용으로 사용되는 예가 없다.

5. GPS 의 경우에는 실내 사용이 어려울 뿐만 아니라 수 cm 이내의 측정 오차를 가지는 경우에는 현 시점에서 만불 이내에서 구매하기가 어렵다.

외란에 영향을 적게 받으면서 사용자의 몸체의 위치와 손목의 위치를 실시간으로 측정할 수 있는 시스템을 제안, 제작 및 분석하는 것이 본 논문의 핵심 내용이 된다. 제스처(gesture)와 연관되는 사용자의 제어 의도는 다양한 방법으로 정의될 수 있으나 본 논문에서는 사람이 동물들을 다룰 때와 같이 직관적인 방법으로써 몸체와 손목의 위치 차를 사용하여 정의하고자 한다.

1. 외란의 영향을 최소화시키며, 자유로운 걸음에 방해가 되지 않으면서 사용자의 몸체와 손목의 위치를 동시에 측정할 수 있는 센서 또는 센서 시스템에 대한 구현 방법은 무엇인가?

2. 측정된 데이터를 이용해서 상술한 기본 기능을 가진 안내 로봇을 제작하기 위해서 로봇 모션 제어 알고리즘을 구현하라.

## 1.2.2 해결 방안

상기한 문제를 해결하기 위해서 본 논문에서는 측정 센서의 측면과 로봇 제어의 측면에서 다음과 같은 방안을 제시한다.

첫번째로 측정 센서의 측면에서는 사용자의 손목과 몸체의 위치를 분리해서 측정한다. 상용 라이다의 경우에 시야각이 50~60 도를 가지므로 몸체와 양 어깨의 위치를 측정하기에 적당하다. 또한 손목의 위치를 측정하기 위해서 시리얼 링크를 이용한 새로운 위치측정 센서인 Intelligent Lead 를 사용하는 것을 제안한다. 본 논문에서 제안하는 Intelligent Lead 는 시리얼 링크의 끝 단에 IMU 를 장착함으로써 손목의 위치 측정뿐만 아니라 LIDAR 로부터 획득되는 위치 데이터를 EKF 로 융합 할 수 있도록 함으로써 로봇 몸체의 흔들림으로 인한 손목의 위치 에러를 최소화시킨다. 다음은 본 논문에서 제안하는 Intelligent Lead 와 사용 예에 관한 개념도를 나타낸다.

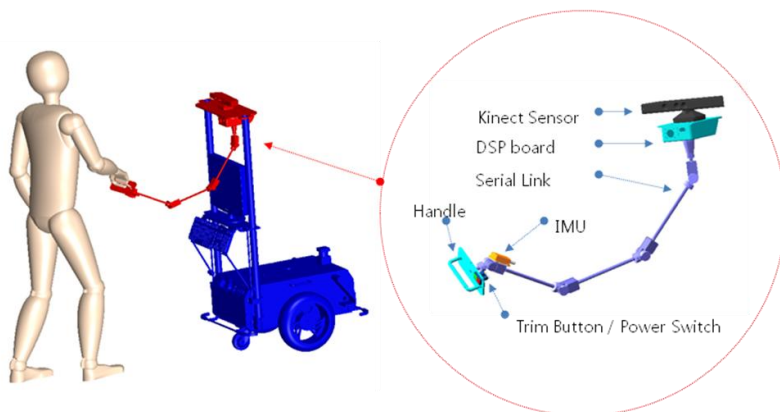


그림 1.1 Intelligent Lead 의 사용 예

LIDAR 만 이용해서 사용자의 관절 데이터를 구하는 경우에는, 사용자가 로봇 가까이 있을 때는 제한된 시야각으로 의해서 사용자의 몸 전체를 감지하기 어려우며 키넥트와 같은 저가형 LIDAR 의 경우에는 측정 정밀도가 많이 떨어지는 문제점이 있다. 또한 LIDAR 와 시리얼 링크의 기준좌표계가 로봇에 붙어 있으므로 거친 지면에 의해서 로봇이 흔들릴 때 측정되는 몸체의 위치, 손목의 위치 및 자세는 로봇의 흔들림의 영향을 그대로 받아서 마치 사용자의 손목이 흔들리는 것으로 인지된다. 따라서 시리얼링크로 손목의 위치 및 자세를 측정하고 LIDAR 로 몸체의 위치를 측정한 후에 시리얼 링크의 끝 단에 붙어서 기구적으로 로봇과 분리되어 있는 IMU(Inertial Measurement Unit)로 로봇의 흔들림에 의한 에러를 보정함으로써 안정적으로 사용자의 위치 및 제어명령을 획득하고자 하는 것이 본 논문의 새로운 접근 방식이 된다.

사용자의 몸체 및 손목의 위치가 실시간으로 구해지면 로봇은 위의 데이터를 바탕으로 상기한 기본 기능들이 구현될 수 있도록 제어 알고리즘이 마련되어야 한다.

두번째로 로봇의 제어기 측면에서는 장애물 회피를 위한 알고리즘으로 GuideCane 에서 사용한 VFH+ 알고리즘을 사용하고 VFH+ 알고리즘을 변형하여 구해진 장애물 회피를 위한 최적의 방향각과 사용자의 손목과 몸체의 위치 차로 인해서 생성되는 사용자의 회전 및 속도 제어 량 및 목표점과의 방향각과 현재의

로봇이 취하는 방향각으로부터 장애물 회피 및 사용자와의 거리 유지뿐만 아니라 사용자의 제어 의도가 반영된 로봇의 제어 알고리즘을 생성한다. 위의 알고리즘은 4 장에 자세히 기술한다.

### 1.3 논문의 구성

본 논문은 다음과 같이 구성된다. 먼저 2 장에서는 제안하는 새로운 위치 측정 시스템의 제작 방법에 대해서 서술한다. 각(angle)과 좌표계(frame)에 대한 정의와 엔코더를 사용하는 시리얼 링크의 구조 및 위치 측정을 위한 관련 수식을 설명한다. 또한 쿼터니언을 사용하는 AHRS 제작 방법에 대해서 서술하며 마지막으로 AHRS 를 이용한 시리얼 링크 제작 방법에 대해서 서술한다.

3 장에서는 현재 널리 사용되고 있는 위치측정시스템에 관해서 정리하며 이들과의 연동을 위한 방법을 서술한다. 먼저 6-DOF 강체의 일반화된 동역학방정식을 서술하며 실외환경에서 비행체의 항법시스템으로 널리 사용되는 GPS/INS integration 방법에 대해서 상술한다. GPS/INS integration 방법을 간략화 시켜서 시리얼 링크와 INS 를 융합하는 방법에 대해서 서술한다. 마지막으로 Intelligent Lead 의 성능 지수를 정의하며 다른 위치 측정 시스템들과의 센서 융합에 적용하는 방법을 서술한다.

4 장에서는 Intelligent Lead 의 출력값 생성 알고리즘과 이동로봇의 모션 제어 알고리즘에 대해서 살펴보고 실험을 위해서 제작된 안내 로봇과 Intelligent Lead 의 하드웨어 구성에 관해서 서술한다.

5 장에서는 4 장에서 제안한 Intelligent Lead 의 출력값 생성 알고리즘과 이동 로봇의 모션 제어 알고리즘을 실험을 통해서 성능 검증한다. 이를 위해서 먼저 센서 출력값을 측정 및 분석하며 장애물 회피 실험 과 모바일 조이스틱 실험, 사용자와 로봇의 협업 실험 및 안내건 모사 실험을 수행한다.

6 장에서는 결과 및 논문의 기여도와 향후 연구 계획에 대해서 서술한다.

## 제 2 장 시리얼 링크를 이용한 위치 측정

이번 장에서는 환경변화에 따른 위치측정시스템의 성능 저하가 중요시 되는 로봇에 적용하기 위한 새로운 위치측정 시스템으로써 시리얼 링크를 이용하는 방법을 제안하며 구현 및 분석을 위해서 필요한 수식을 서술한다. 시리얼 링크는 강체로 형성된 링크와 베어링으로 형성된 관절이 직렬로 연결된 형태이기 때문에 비전 기반의 센서들이 가지는 연무나 먼지 등에 의해서 성능저하가 일어나지 않으며 시리얼 링크 사이에 순간적인 끼어듦으로 인해서 영상의 단절이 생기더라도 출력 값의 영향을 받지 않는다. 관절각을 측정하는 방법과 링크각을 측정하는 방법이 있으며, 각각에 대해서 관련된 수식을 정리한다. 본 논문에서 다루는 시리얼 링크는 한쪽은 로봇에 부착되고 다른 한쪽은 손으로 잡게 되므로 자유로운 운동이 가능한 구조를 가진다. 따라서 로보틱스(robotics) 분야에서의 기구학이나 동역학뿐만 아니라 항공학(aeronautics)에서의 6D 강체 동역학의 이해가 필수적이다. 그러나 동체나 링크지의 형태에 따른 aero dynamic drag 와 같은 공기역학에 관해서 세부적인 요소(factor)들을 고려하지는 않는다. 본 장에서는 서로 다르게 표현되는 표기법에 의한 수식 이해의 혼란을 줄이기 위해서 먼저 기본적인 각과 좌표계를 정의한다.



## 2.1 좌표계와 회전 행렬 정의

로보틱스 분야와 항공학 분야에서는 각을 어디서 측정하는가에 따라서 회전 각도의 정의에서 차이를 보인다. 롤(roll), 피치(pitch), 요우(yaw)각을 정의하는 데에 있어서 항공기내에서 각을 측정하는 경우와 사용자가 바라본 시각에서 로봇의 각을 측정한다는 관점상의 차이가 있다. 또한 롤/피치/요우 회전축이 X/Y/Z 축으로 정의되는 경우와 Z/Y/X 축으로 정의되는 경우가 있어서 다소 혼란스러움을 초래한다.

아래의 표는 이러한 차이를 정리한 것이다.

표 2.1 롤/피치/요우 회전 축과 기본 회�행렬 정의

	Aeronautics	Robotics
각 측정 기준 프레임	Body Frame	Base (Inertial) Frame
Basic Rotation Matrix	$R_x(\theta) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \theta & \sin \theta \\ 0 & -\sin \theta & \cos \theta \end{bmatrix}$ $R_y(\theta) = \begin{bmatrix} \cos \theta & 0 & -\sin \theta \\ 0 & 1 & 0 \\ \sin \theta & 0 & \cos \theta \end{bmatrix}$ $R_z(\theta) = \begin{bmatrix} \cos \theta & \sin \theta & 0 \\ -\sin \theta & \cos \theta & 0 \\ 0 & 0 & 1 \end{bmatrix}$	$R_x(\theta) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \theta & -\sin \theta \\ 0 & \sin \theta & \cos \theta \end{bmatrix}$ $R_y(\theta) = \begin{bmatrix} \cos \theta & 0 & \sin \theta \\ 0 & 1 & 0 \\ -\sin \theta & 0 & \cos \theta \end{bmatrix}$ $R_z(\theta) = \begin{bmatrix} \cos \theta & -\sin \theta & 0 \\ \sin \theta & \cos \theta & 0 \\ 0 & 0 & 1 \end{bmatrix}$
Axis(Roll/Pitch/Yaw)	X,Y,Z	Z,Y,X (Fu) / X,Y,Z (Craig)

회전축의 순서에 따라서 다양한 회�행렬(rotation matrix)가 정의된다. 항공학분야(aeronautics)에서는 NASA standard airplane과 NASA standard aerospace가 쓰이며 로보틱스 분야에서는 ZYX

롤/피치/요우 각과 ZXZ Euler 각, ZYZ Euler 각이 주로 쓰이며 각각에 대해서 정리하면 다음과 같다.

표 2.2 대표적인 DCM들

<p>항공학분야</p>	<p>1. NASA standard Airplane</p> $R_x(\phi)R_y(\theta)R_z(\psi) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\phi & \sin\phi \\ 0 & -\sin\phi & \cos\phi \end{bmatrix} \begin{bmatrix} \cos\theta & 0 & -\sin\theta \\ 0 & 1 & 0 \\ \sin\theta & 0 & \cos\theta \end{bmatrix} \begin{bmatrix} \cos\psi & \sin\psi & 0 \\ -\sin\psi & \cos\psi & 0 \\ 0 & 0 & 1 \end{bmatrix}$ $= \begin{bmatrix} \cos\theta\cos\psi & \cos\theta\sin\psi & -\sin\theta \\ \sin\phi\sin\theta\cos\psi - \cos\phi\sin\psi & \sin\phi\sin\theta\sin\psi + \cos\phi\cos\psi & \sin\phi\cos\theta \\ \cos\phi\sin\theta\cos\psi + \sin\phi\sin\psi & \cos\phi\sin\theta\sin\psi - \sin\phi\cos\psi & \cos\phi\cos\theta \end{bmatrix}$ <p>2. NASA standard Aerospace</p> $R_z(\phi)R_y(\theta)R_x(\psi) = \begin{bmatrix} \cos\phi & \sin\phi & 0 \\ -\sin\phi & \cos\phi & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos\theta & 0 & -\sin\theta \\ 0 & 1 & 0 \\ \sin\theta & 0 & \cos\theta \end{bmatrix} \begin{bmatrix} \cos\psi & \sin\psi & 0 \\ -\sin\psi & \cos\psi & 0 \\ 0 & 0 & 1 \end{bmatrix}$ $= \begin{bmatrix} \cos\phi\cos\theta\cos\psi - \sin\phi\sin\psi & \cos\phi\cos\theta\sin\psi + \sin\phi\cos\psi & -\cos\phi\sin\theta \\ -\sin\phi\cos\theta\cos\psi - \cos\phi\sin\psi & -\sin\phi\cos\theta\sin\psi + \cos\phi\cos\psi & \sin\phi\sin\theta \\ \sin\theta\cos\psi & \sin\theta\sin\psi & \cos\theta \end{bmatrix}$
<p>로보틱스분야</p>	<p>1. Roll/Pitch/Yaw Angle</p> $R_z(\phi)R_y(\theta)R_x(\psi) = \begin{bmatrix} \cos\phi & -\sin\phi & 0 \\ \sin\phi & \cos\phi & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos\theta & 0 & \sin\theta \\ 0 & 1 & 0 \\ -\sin\theta & 0 & \cos\theta \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\psi & -\sin\psi \\ 0 & \sin\psi & \cos\psi \end{bmatrix}$ $= \begin{bmatrix} \cos\phi\cos\theta & \cos\phi\sin\theta\sin\psi - \sin\phi\cos\psi & \cos\phi\sin\theta\cos\psi + \sin\phi\sin\psi \\ \sin\phi\cos\theta & \sin\phi\sin\theta\sin\psi + \cos\phi\cos\psi & \sin\phi\sin\theta\cos\psi - \cos\phi\sin\psi \\ -\sin\theta & \cos\theta\sin\psi & \cos\theta\cos\psi \end{bmatrix}$ <p>2. Eulerian Angle I</p> $R_z(\phi)R_x(\theta)R_z(\psi) = \begin{bmatrix} \cos\phi & -\sin\phi & 0 \\ \sin\phi & \cos\phi & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\theta & -\sin\theta \\ 0 & \sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} \cos\psi & -\sin\psi & 0 \\ \sin\psi & \cos\psi & 0 \\ 0 & 0 & 1 \end{bmatrix}$ $= \begin{bmatrix} \cos\phi\cos\psi - \sin\phi\cos\theta\sin\psi & -\cos\phi\sin\psi - \sin\phi\cos\theta\cos\psi & \sin\phi\sin\theta \\ \sin\phi\cos\psi + \cos\phi\cos\theta\sin\psi & -\sin\phi\sin\psi + \cos\phi\cos\theta\cos\psi & -\cos\phi\sin\theta \\ \sin\theta\sin\psi & \sin\theta\cos\psi & \cos\theta \end{bmatrix}$ <p>3. Eulerian Angle II</p> $R_z(\phi)R_y(\theta)R_z(\psi) = \begin{bmatrix} \cos\phi & -\sin\phi & 0 \\ \sin\phi & \cos\phi & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos\theta & 0 & \sin\theta \\ 0 & 1 & 0 \\ -\sin\theta & 0 & \cos\theta \end{bmatrix} \begin{bmatrix} \cos\psi & -\sin\psi & 0 \\ \sin\psi & \cos\psi & 0 \\ 0 & 0 & 1 \end{bmatrix}$ $= \begin{bmatrix} \cos\phi\cos\theta\cos\psi - \sin\phi\sin\psi & -\cos\phi\cos\theta\sin\psi - \sin\phi\cos\psi & \cos\phi\sin\theta \\ \sin\phi\cos\theta\cos\psi + \cos\phi\sin\psi & -\sin\phi\cos\theta\sin\psi + \cos\phi\cos\psi & \sin\phi\sin\theta \\ -\sin\theta\cos\psi & \sin\theta\sin\psi & \cos\theta \end{bmatrix}$

이후로 본 논문에서는 이러한 좌표계 변환을 위한 회전행렬을 DCM(Direction Cosine Matrix)으로 표기한다.

회전 변환 행렬을 정의할 때 피치각이 특정 값에서 특이점(singularity)을 가진다. 이로 인해서 적용 시스템에 따라서 선택적으로 사용되어야만 한다. Pitch 각이 0도 근방에서 주로 사용되는 비행체의 경우에는  $\pm 90^\circ$ 에서 특이점이 생기는 NASA standard airplane을 사용하며, pitch 각이 90도 근방에서 주로 사용되는 위성의 경우에는 특이점이 0도에 위치해 있는 NASA standard aerospace를 사용한다. [20, 21]

로보틱스 분야에서 정의되는 롤/피치/요우 각도를 NASA standard airplane과 회전 축 표기를 동일하게 하기 위해서  $\phi$  와  $\psi$  를 바꾸면 서로가 전치(transpose) 관계에 있는 것을 알 수 있다. 이것은 롤/피치/요우 각이 동체좌표계에서 관성좌표계로의 회전변환행렬로 정의되고, NASA standard airplane은 관성좌표계에서 동체좌표계로의 회전변환행렬로 정의되기 때문이다. 이러한 혼동을 없애기 위해서 좌표계 사이의 관계를 의미하는 회전행렬을 표시할 때 위 첨자와 아래첨자에 좌표계 기호를 표기해서 아래 첨자 좌표계에서 위 첨자 좌표계으로의 변환 행렬로 나타낸다.

표 2.3 DCM의 수식적 관계

In the Aeronautics	In the Robotics
<p>1. NASA Standard Airplane</p> $C_i^c = R_x(\phi)R_y(\theta)R_z(\psi) = \begin{bmatrix} 1 & 0 & 0 & \cos\theta & 0 & -\sin\theta & \cos\psi & \sin\psi & 0 \\ 0 & \cos\phi & \sin\phi & 0 & 1 & 0 & -\sin\psi & \cos\psi & 0 \\ 0 & -\sin\phi & \cos\phi & \sin\theta & 0 & \cos\theta & 0 & 0 & 1 \end{bmatrix}$ $= \begin{bmatrix} \cos\phi\cos\psi & \cos\phi\sin\psi & -\sin\theta \\ \sin\phi\sin\theta\cos\psi - \cos\phi\sin\psi & \sin\phi\sin\theta\sin\psi + \cos\phi\cos\psi & \sin\phi\cos\theta \\ \cos\phi\sin\theta\cos\psi + \sin\phi\sin\psi & \cos\phi\sin\theta\sin\psi - \sin\phi\cos\psi & \cos\phi\cos\theta \end{bmatrix}$	<p>1. Roll/Pitch/Yaw Angle</p> $C_i^c = R_x(\phi)R_y(\theta)R_z(\psi) = \begin{bmatrix} \cos\phi & -\sin\phi & 0 & \cos\theta & 0 & \sin\theta & 1 & 0 & 0 \\ \sin\phi & \cos\phi & 0 & 0 & 1 & 0 & 0 & \cos\psi & -\sin\psi \\ 0 & 0 & 1 & -\sin\theta & 0 & \cos\theta & 0 & \sin\psi & \cos\psi \end{bmatrix}$ $= \begin{bmatrix} \cos\phi\cos\theta & \cos\phi\sin\theta\sin\psi - \sin\phi\cos\psi & \cos\phi\sin\theta\cos\psi + \sin\phi\sin\psi \\ \sin\phi\cos\theta & \sin\phi\sin\theta\sin\psi + \cos\phi\cos\psi & \sin\phi\sin\theta\cos\psi - \cos\phi\sin\psi \\ -\sin\theta & \cos\theta\sin\psi & \cos\theta\cos\psi \end{bmatrix}$
$C_i^b = (C_i^c)^T$	
	<p>2. If Convert Roll/Yaw - axis from Z/X - axis to X/Z - axis</p> $C_i^c = R_x(\psi)R_y(\theta)R_z(\phi) = \begin{bmatrix} \cos\psi & -\sin\psi & 0 & \cos\theta & 0 & \sin\theta & 1 & 0 & 0 \\ \sin\psi & \cos\psi & 0 & 0 & 1 & 0 & 0 & \cos\phi & -\sin\phi \\ 0 & 0 & 1 & -\sin\theta & 0 & \cos\theta & 0 & \sin\phi & \cos\phi \end{bmatrix}$ $= \begin{bmatrix} \cos\theta\cos\psi & \sin\phi\sin\theta\cos\psi - \cos\phi\sin\psi & \cos\phi\sin\theta\cos\psi + \sin\phi\sin\psi \\ \cos\theta\sin\psi & \sin\phi\sin\theta\sin\psi + \cos\phi\cos\psi & \cos\phi\sin\theta\sin\psi - \sin\phi\cos\psi \\ -\sin\theta & \sin\phi\cos\theta & \cos\phi\cos\theta \end{bmatrix}$

또한 측정 좌표계는 수식의 이해에 혼란을 가져올 때만 표기하며 의미상 명확한 것에 대해서는 표기하지 않는다. 예를 들어 회전 각속도를 나타내는  $\omega$ 는 동체좌표계의 자이로스코프 센서를 통해서 획득되므로  $\omega_b$ 의 의미를 가진다.

본 논문에서는 특이점이  $\pm 90^\circ$ 에 위치하는 NASA standard airplane을 사용한다. 또한 롤/피치/요우 각도는  $[\phi \ \theta \ \psi]$ 로 표기하고 X/Y/Z축에서의 회전각으로 정의하며 각각의 각들은 항법좌표계에서 본 동체좌표계의 각으로 정의한다.

$$C_n^b = \begin{bmatrix} \cos\theta\cos\psi & \cos\theta\sin\psi & -\sin\theta \\ (\sin\phi\sin\theta\cos\psi - \cos\phi\sin\psi) & (\sin\phi\sin\theta\sin\psi + \cos\phi\cos\psi) & \sin\phi\cos\theta \\ (\cos\phi\sin\theta\cos\psi + \sin\phi\sin\psi) & (\cos\phi\sin\theta\sin\psi - \sin\phi\cos\psi) & \cos\phi\cos\theta \end{bmatrix} \quad (2.1)$$

위의 수식에서 첨자 n은 항법좌표계를 의미하며 b는 동체좌표계를 의미한다. 본 논문에서는 이와 함께 첨자 i를 관성좌표계를 의미할 때 사용한다. 항법좌표계는 지표면 위에 설정되어서 관성좌표계에서

고려하는 지구회전속도를 고려하지 않는 좌표계가 된다.

로봇의 링크와 관절각에 대한 관계를 정의하는 D-H(Denavit-Hartenberg) 표기법은 로봇틱스 교재마다 표기가 상이하다. 서로 간에 역(inverse)나 전치의 관계에 있지 않기 때문에 수식 전개 전에 어느 표기를 따를 것인 지가 미리 정의되어 있어야만 한다. 아래의 그림은 Fu와 Craig 교재의 동차변환행렬(homogeneous transformation matrix)을 나타낸다.

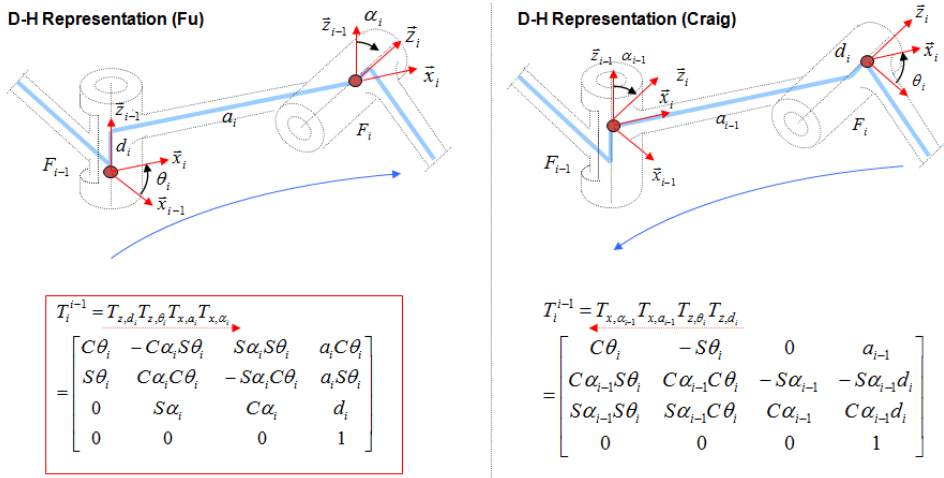


그림 2.1 동차변환행렬의 비교

본 논문에서는 Fu 교재의 표기를 따른다.[22] 위의 그림에서 빨간색 박스로 표시된 수식이 향후 본 논문 전개에서 사용될 동차변환행렬이 된다.

## 2.2 쿼터니언에 기반한 AHRS

시리얼 링크의 각 끝 단의 위치 차를 구하기 위해서 앞에서 서술한 관절각을 구하고 동차변환행렬들의 곱을 구하는 방법 외에 관성좌표계에서 본 각각의 링크들의 3 차원 각도를 측정하고 각각의 링크마다 회전변환행렬을 구해서 위치 차를 구하는 방법을 생각할 수 있다. 본 논문에서는 시리얼 링크의 각 링크들에 AHRS 를 달아서 링크각을 측정하고 위의 데이터를 이용해서 시리얼 링크의 각 끝단 사이의 위치 차를 구하고자 한다. 따라서 소형화된 AHRS 의 제작이 필요하다.

아래의 식 (2.2)는 동체좌표계에서 측정한 회전속도 벡터를 관성좌표계에서의 오일러 각의 미분 값으로 변환하기 위한 관계식을 나타낸다.

$$\begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} 1 & \sin \phi \tan \theta & \cos \phi \tan \theta \\ 0 & \cos \phi & -\sin \phi \\ 0 & \frac{\sin \phi}{\cos \theta} & \frac{\cos \phi}{\cos \theta} \end{bmatrix} \begin{bmatrix} p \\ q \\ r \end{bmatrix} \quad (2.2)$$

위의 수식에서  $\cos \Theta$  가 0 이 되는  $\Theta$  값에서 특이점이 발생한다. 즉, 구동기(manipulator)의 관점에서 보면 오일러 각의 미소 변화시키기 위해서는 아주 큰 동체좌표계의 회전 속도의 변화를 필요로 하며 측정기(sensor) 관점에서 보면 특이점 가까이에서 미소한 동체좌표계의 회전 속도 측정 에러가 오일러 각을 연산하는데 있어서 큰 연산 에러를 유발시킨다. 쿼터니언은 이러한 소위

gimbal-lock 이라고 불리우는 고질적인 특이점 문제(long-standing problem of singularity)를 해결하기 위해서 항공학 분야에서 많이 사용하고 있으며, 본 논문에서도 연산과정에서 오일러 각 대신에 쿼터니언을 사용하여 물체의 자세를 표기한다.

쿼터니언은 아래의 추상적인 수학적 정의에 바탕을 둔다.

$$\begin{aligned} Q &= q_0 + q_1\bar{i} + q_2\bar{j} + q_3\bar{k}, Q^* = q_0 - q_1\bar{i} - q_2\bar{j} - q_3\bar{k} \\ \bar{i}^2 &= \bar{j}^2 = \bar{k}^2 = \bar{i}\bar{j}\bar{k} = -1 \\ \bar{i}\bar{j} &= \bar{k} = -\bar{j}\bar{i}, \bar{j}\bar{k} = \bar{i} = -\bar{k}\bar{j}, \bar{k}\bar{i} = \bar{j} = -\bar{i}\bar{k} \end{aligned} \quad (2.3)$$

단위 크기의 쿼터니언은 아래의 식을 만족시키는 유일한 해  $\theta$ 를 가진다

$$Q = \cos \frac{\theta}{2} + \bar{u} \sin \frac{\theta}{2}, \bar{u} = \frac{Q}{\|Q\|} \quad (2.4)$$

임의의 3 차원 벡터,  $v$  를 실수부가 0 인 쿼터니언,  $v_q$  로 대응시킨 후에 Linear operator,  $L_q(v_q)$ 를 다음과 같이 정의한다.

$$L_q(v_q) = Qv_qQ^* \quad (2.5)$$

식 (2.5)에 의해서 생성되는 새로운 쿼터니언의 허수부,  $L(v_q)$ 는 3 차원 좌표계에서  $v$  벡터를 쿼터니언의 허수부가 이루는 벡터를 기준으로  $\theta$ 만큼 회전 이동해서 생성된 벡터를 나타낸다.

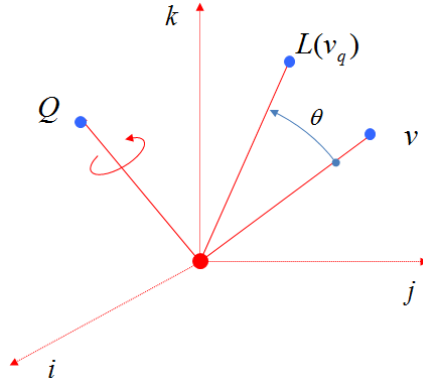


그림 2.2 쿼터니언 회전 연산자

따라서 회�행렬을 곱하는 과정 보다 연산량이 줄어들 뿐만 아니라 아래의 수식 (2.6)처럼 동체좌표계의 회전속도를 쿼터니언의 미분값으로 변환하는 수식에서 특이점이 발생하지 않게 된다.

$$\dot{Q} = \frac{1}{2} Q \otimes \Omega = \frac{1}{2} \begin{bmatrix} q_0 & -q_1 & -q_2 & -q_3 \\ q_1 & q_0 & -q_3 & q_2 \\ q_2 & q_3 & q_0 & -q_1 \\ q_3 & -q_2 & q_1 & q_0 \end{bmatrix} \begin{bmatrix} 0 \\ p \\ q \\ r \end{bmatrix} \quad (2.6)$$

위의 수식에서  $\otimes$  는 쿼터니언 벡터의 외적(cross product)을 의미한다. 또한  $\Omega$  는 동체좌표계의 회전속도벡터를 실수부가 0인 쿼터니언으로 변환된 값을 벡터화 한 것이다. 그러나 쿼터니언은 그 자체로써 물리적인 의미를 가지지 못하므로 연산은 쿼터니언에 의해서 이루어지며 최종 출력은 오일러 각으로 나타내는 것이



일반적이다. 이를 위해서 필요로 하는 쿼터니언과 오일러 각 사이의 관계식을 아래의 표로 나타냈다.

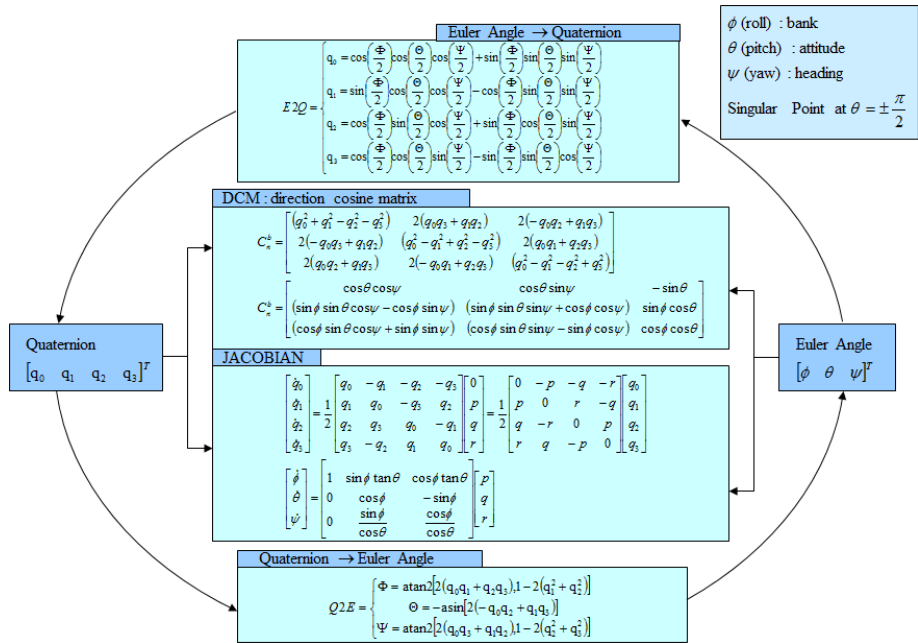


그림 2.3 쿼터니언과 오일러 각의 상호 변환 관계식

## 2.3 관절각 측정 방식의 Intelligent Lead

Intelligent Lead 의 시리얼 링크의 한쪽에는 IMU 가 부착되어 있으며 안내 로봇에 탑재된 LIDAR 와 연동하여 사용된다.

시리얼 링크는 회전형 엔코더 또는 절대각 엔코더들을 연속적으로 링크들과 연결한 형태가 된다. 이때 사용하는 엔코더는 정밀하면서도 소형인 것을 택해야 하지만 일반적으로 정밀할 수록 크기가 커지게 된다. 본 논문에서는 usdigital 사의 MAE2-A2 을

사용했다. 위의 엔코더는 자기형(magnetic) 절대각 엔코더로써 360 도 회전 시에 4096 의 정밀도(resolution)를 가진 PWM 펄스를 내보낸다. 따라서 각각의 관절각이 0.088 도의 정밀도를 가진다. 또한 PWM 펄스의 주기가 4ms 로써 250Hz 의 샘플링률(sampling rate)을 가진다. 아래의 그림은 제작된 시리얼 링크와 실험에서 사용한 엔코더를 나타낸다.

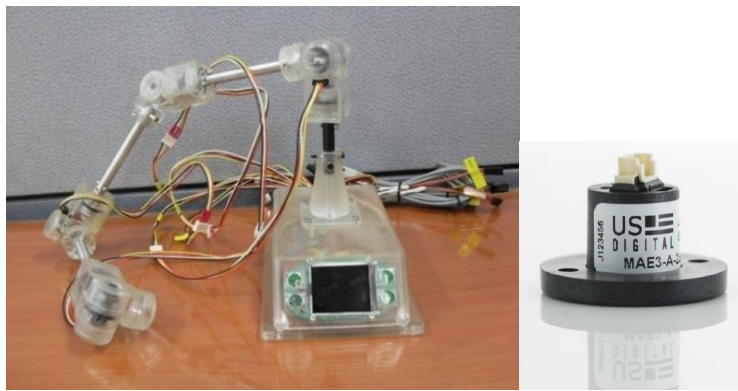
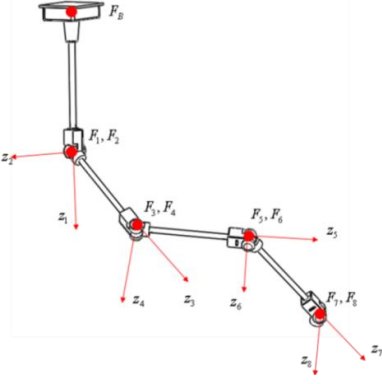


그림 2.4 시리얼 링크와 사용된 절대각 엔코더

시리얼 링크는 여유자유도(redundancy)를 가지는 8 회전축으로 구성되며 각각의 회전축에 절대각 엔코더를 부착함으로써 회전형(Incremental) 엔코더를 부착했을 때 필요로 하는 초기 정렬 과정 없이 사용 가능하도록 설계했다.

본 논문에서 실험을 위해 제작된 시리얼 링크의 D-H 인자를 표로 나타내면 아래와 같다.

표 2.4 시리얼 링크의 D-H 인자들

좌표계 and Transformation Matrix	$\theta$ (deg.)	$\alpha$ (deg.)	$a$ (mm)	$d$ (mm)
	$T_1^B$	$\theta_1$	0	50
	$T_2^1$	$\theta_2$	0	0
	$T_3^2$	$\theta_3$	-90	200
	$T_4^3$	$\theta_4$	90	0
	$T_5^4$	$\theta_5$	-90	200
	$T_6^5$	$\theta_6$	90	0
	$T_7^6$	$\theta_7$	-90	200
	$T_8^7$	$\theta_8$	90	0
	$T_R^8$	0	-90	10

$T_S^8$  을 IMU 가 부착된 센서 좌표계로부터 끝 단까지의 변환 행렬이라 하면, 센서 좌표계로부터 기준좌표계로의 변환 행렬은 다음과 같이 계산한다.

$$T_S^B = T_1^B T_2^1 T_3^2 \dots T_8^7 T_S^8 \quad (2.7)$$

식 (2.7)에서 구해진  $T_S^B$  값의  $i$  행,  $j$  열의 값을  $T[i,j]$ 로 정의하면, 기준좌표계에 대한 센서 좌표계의 상대적인 위치 및 자세(pose)는 다음과 같이 구해진다.

$$P_x = T[1,4], P_y = T[2,4], P_z = T[3,4] \quad (2.8)$$

$$\theta = \tan^{-1} \left( \frac{-T[3,1]}{\sqrt{T[3,2]^2 + T[3,3]^2}} \right), \psi = \tan^{-1} \left( \frac{T[2,1]}{T[1,1]} \right), \phi = \tan^{-1} \left( \frac{T[3,2]}{T[3,3]} \right) \quad (2.9)$$

시리얼 링크의 끝 단이 연결된 로봇 몸체에 사용자의 어깨 및 몸체의 위치를 측정하기 위해서 키넥트 센서를 부착했다. 키넥트는 적외선 패턴을 물체에 투영하고 적외선 센서로 패턴(pattern)을 측정해 깊이 정보를 획득한다. 일반 카메라를 사용하는 것에 비해 주변광(Ambient Light)의 간섭이 적고, 밤에도 사용이 가능하다는 장점이 있다. RGB 카메라도 함께 장착되어 있기 때문에 특정 픽셀에서의 깊이 정보와 RGB 로 이루어진 RGBD 4 채널 데이터를 제공한다. 본 논문에서는 사용자의 관절 정보는 획득하기 위해서 키넥트 센서와 OpenNI 소프트웨어를 사용했다. 로봇이 거친 지면을 지날 때나 급격한 움직임에 의해서 시리얼 링크의 기준좌표계가 흔들리는 경우에 사용자의 손목에 대한 좌표는 위의 진동이 그대로 반영된다. 따라서 로봇의 기준좌표계의 진동을 보상하기 위해서 사용자의 손이 잡히는 부분에 IMU 를 장착하고 측정되는 가속도센서(accelerometer)와 자이로스코프(gyroscope) 데이터를 EKF 의 Time prediction 절차에 사용한다. 실험을 위해서 사용된 IMU 센서는 xsens 사의 Mti 센서로 100Hz 의 측정 샘플링 률을 가진다. 아래의 그림은 사용된 센서들을 나타낸다.



그림 2.5 사용된 IMU 센서 와 키넥트 센서

Intelligent Lead 를 제작하기 위해서 사용되는 센서들은 다음과 같은 측정 에러 요소를 가지고 있다. 먼저 저가형 IMU 는 자이로스코프의 드리프트가 누적 에러를 발생시키며 시리얼 링크에 사용되는 엔코더 또한 무게를 줄이기 위해서 소형제품을 사용할수록 엔코더 디스크 판이 작아짐으로써 정밀도가 떨어진다. 또한 시리얼 링크는 링크의 길이가 길어질수록 위치 및 자세에러가 비례적으로 증가하게 된다. 링크의 길이나 엔코더의 정밀도뿐 만 아니라 조립 정밀도에 따라서도 위치 및 자세 측정 에러가 달라지게 된다. 상술한 것처럼 LIDAR 센서는 연무나 장애물에 의해서 끼어듦이 생기거나 동적 환경에서는 성능저하를 일으킨다. 따라서 이들 센서는 단독적으로 사용하기보다는 융합을 통해서 통합시스템의 에러를 줄이는 것이 필요하다. 이러한 통합은 정밀도 향상뿐만 아니라 센서들 중 하나가 고장 났을 때 또는 사용자가 시리얼 링크를 놓치거나 로봇과 사용자 사이에 순간적인 끼어듦이 생겼을 때에도 출력 데이터가 급격히 튀는 것을 방지한다.

## 2.4 링크각 측정 방식의 Intelligent Lead

MEMS 기술의 발전은 항공전자 센서의 발전을 가져왔다. 특히 관성 센서는 스마트 폰에 들어갈 정도로 소형, 경량화 되었으며 가격 또한 낮아지게 되었다.

MEMS 센서를 이용해서 3차원 자세를 구하는 연구는 많은 선행 연구자에 의해서 수행되어 왔다. 또한 MEMS 센서를 이용한 모션캡처(motion capture) 장비가 Xsens 사를 비롯한 몇몇 회사에서 상품화 되고 있다.

이전 장에서는 절대각 엔코더로 관절각을 측정하고 순기구학을 풀어서 시리얼 링크의 위치 차를 구했었다. 이번 장에서는 시리얼 링크를 사용자의 사용 편리성을 증대시키기 위해서 작고 가볍게 제작하기 위한 새로운 방법을 제안한다. 엔코더들의 결합으로 이루어진 시리얼 링크는 엔코더의 크기로 인해서 작게 만들 수 없는 단점이 있다. 또한 엔코더로 구성된 관절은 한 축에 한 방향의 자유도를 가지므로 제한된 자유도를 가지는 여러 개의 관절들이 모여진 시리얼 링크는 많은 특이점을 가지게 된다. 본 논문에서는 이러한 단점을 줄이기 위해서 구형 관절로 연결된 시리얼 링크를 제안한다. 시리얼 링크의 각각의 링크마다 3축 자이로스코프, 3축 가속도 센서 와 3축 지자기 센서(magnetometer)를 부착해서 링크의 각들을 구하고 구해진 링크각들을 이용해서 양 끝 단의 위치 차를 구한다.

아래의 그림은 실험용으로 제작된 AHRS 기반의 Intelligent Lead 를 나타낸다.

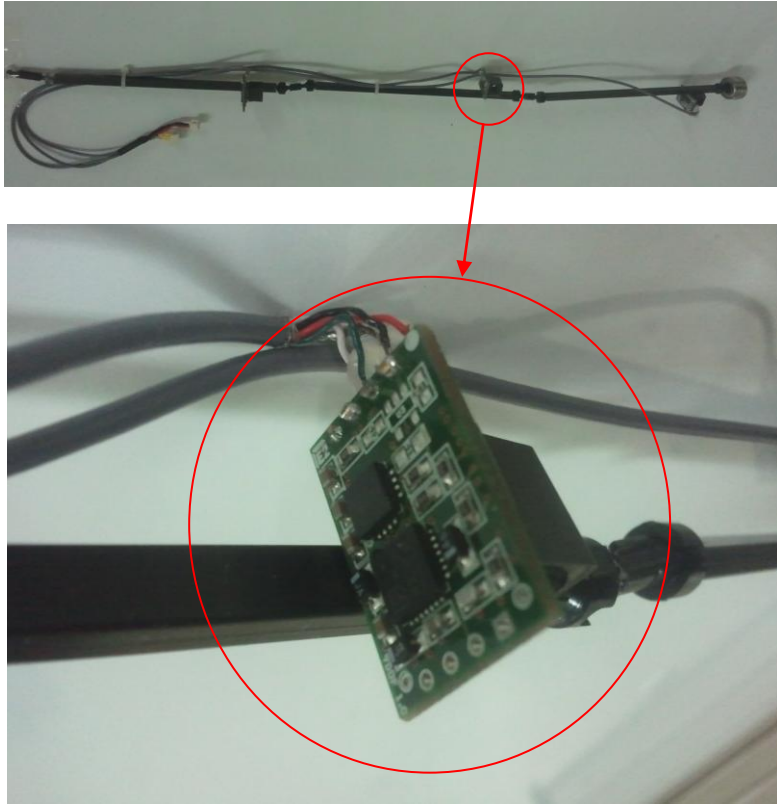


그림 2.6 링크각 측정 방식의 Intelligent Lead

MEMS센서는 ST Micron사의 자이로스코프, 가속도센서와 지자기센서를 사용했다. 프로세서로는 ST Micron사의 STM-32 chip을 사용했다.

앞 장에서 서술한 식(2.6)을 이용해서 상태변수(state)를 갱신하고 가속도센서와 마그네틱 센서 값으로부터 EKF 의 측정값 갱신하는 방법이 사용된다. 아래의 표는 EKF 와 쿼터니언 관계식을 이용한 AHRS 제작에 필요한 수식을 정리한 것이다.

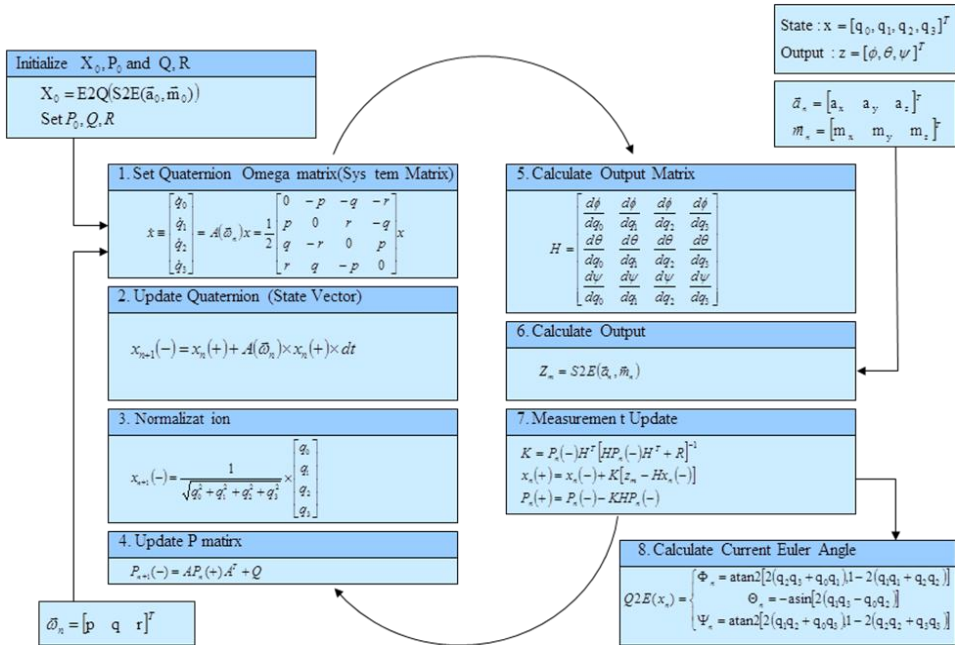


그림 2.7 EKF 로 구현된 AHRS 블록다이어그램

위의 표에서 출력행렬(output matrix)과 가속도센서와 자자기센서 데이터로부터 오일러각을 구하는 함수인 S2E 는 표의 공간이 부족해서 수식을 나타내지 않았다.

출력행렬을 구하기 위해서 다음의 상수들을 정의한다.

$$\begin{aligned}
 C_r &= \frac{2}{DCM(1,2)^2 + DCM(2,2)^2} \\
 C_p &= \frac{-1}{\sqrt{1 - DCM(0,2)^2}} \\
 C_y &= \frac{2}{DCM(0,0)^2 + DCM(0,1)^2}
 \end{aligned} \tag{2.10}$$

수식 (2.10)에서 DCM(a,b)는 앞의 그림 2.3 에서 쿼터니언으로 표기된 DCM 행렬의 a 행 b 열의 원소를 나타낸다. 정의된 상수들과



DCM 의 원소들을 이용해서 출력행렬을 이루는 편미분 값들을 나타내면 다음과 같다.

$$\begin{aligned}
 \frac{d\phi}{dq_0} &= C_r [q_1 DCM(2,2)] \\
 \frac{d\phi}{dq_1} &= C_r [q_0 DCM(2,2) - DCM(1,2) \times (-2q_1)] \\
 \frac{d\phi}{dq_2} &= C_r [q_3 DCM(2,2) - DCM(1,2) \times (-2q_2)] \\
 \frac{d\phi}{dq_3} &= C_r [q_2 DCM(2,2)] \\
 \frac{d\theta}{dq_0} &= C_p (-2q_2) \\
 \frac{d\theta}{dq_1} &= C_p (2q_3) \\
 \frac{d\psi}{dq_2} &= C_p (-2q_0) \\
 \frac{d\theta}{dq_3} &= C_p (2q_1) \\
 \frac{d\psi}{dq_0} &= C_y [q_3 DCM(0,0)] \\
 \frac{d\psi}{dq_1} &= C_y [q_2 DCM(0,0)] \\
 \frac{d\psi}{dq_2} &= C_y [q_1 DCM(0,0) - DCM(0,1) \times (-2q_2)] \\
 \frac{d\psi}{dq_3} &= C_y [q_0 DCM(0,0) - DCM(0,1) \times (-2q_3)]
 \end{aligned} \tag{2.11}$$

식 (2.11)은 쿼터니언으로 표기된 DCM 을 쿼터니언 원소들로 편미분해서 구해진 것이다.

가속도센서와 지자기센서 데이터로부터 오일러 각을 구하는 절차는 먼저 가속도 센서의 데이터로부터 중력방향에 대한 롤 값과 피치 값을 다음과 같이 구한다.

동체좌표계에서 측정되는 중력은 다음과 같다.

$$f^b = C_n^b f^n = C_n^b \begin{bmatrix} 0 \\ 0 \\ -g \end{bmatrix} = \begin{bmatrix} g \sin \theta \\ -g \sin \phi \cos \theta \\ -g \cos \phi \cos \theta \end{bmatrix} = \begin{bmatrix} f_x \\ f_y \\ f_z \end{bmatrix} \quad (2.12)$$

식 (2.12)에서  $C_n^b$  는 그림 2.3 에서 오일러각으로 표기된 DCM 을 나타낸다. 동체좌표계에 부착된 가속도센서의 값이  $f^b = [f_x \ f_y \ f_z]^T$  이면 롤과 피치값은 식 (2.12)로부터 다음과 같이 구해진다.

$$\begin{aligned} \phi &= \tan^{-1} \left( \frac{f_y}{f_z} \right) \\ \theta &= \tan^{-1} \left( \frac{f_x}{\sqrt{f_y^2 + f_z^2}} \right) \end{aligned} \quad (2.13)$$

같은 방법으로 자북의 방향을 x 축으로 잡고 지자기 센서의 값을 정규화시키는 것으로 가정하면 동체좌표계에서 측정되는 지자기 힘은 다음과 같다.

$$h^b = C_n^b h^n = C_n^b \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} \cos \theta \cos \psi \\ \sin \phi \sin \theta \cos \psi - \cos \phi \sin \psi \\ \cos \phi \sin \theta \cos \psi + \sin \phi \sin \psi \end{bmatrix} = \begin{bmatrix} h_x \\ h_y \\ h_z \end{bmatrix} \quad (2.14)$$

롤과 피치가 구해지면 요우값은 지자기센서에서 측정된 값으로부터 구해진다. 정규화된 지자기 센서의 값이  $h^b = [h_x \ h_y \ h_z]^T$  이면 요우 값은 식 (2.14)에 의해서 다음과 같이 구해진다.

$$\psi = \cos^{-1}\left(\frac{h_x}{\cos \theta}\right) \quad (2.15)$$

관성좌표계에서 측정한 각각의 링크의 쿼터니언이나 오일러 각을 구한 후에 시리얼 링크의 끝 단의 위치 차를 다음과 같이 구할 수 있다.

각각의 길이가  $d_k$  인  $m$  개의 링크가 구형 관절로 연결되어 있을 때 기준좌표계에 대한 끝 단의 위치 벡터  $P$  는 다음과 같이 구해진다.

$$P = \sum_{k=1}^m C_b^i(Q) [d_k \quad 0 \quad 0]^T \quad (2.16)$$

쿼터니언을 기반으로 위치차를 구한다면 식 2.16 에서 회전행렬은 다음과 같다.

$$C_i^b(Q) = \begin{bmatrix} 1 - 2(q_2^2 + q_3^2) & 2(q_0q_3 + q_1q_2) & 2(-q_0q_2 + q_1q_3) \\ 2(-q_0q_3 + q_1q_2) & 1 - 2(q_1^2 + q_3^2) & 2(q_0q_1 + q_2q_3) \\ 2(q_0q_2 + q_1q_3) & 2(-q_0q_1 + q_2q_3) & 1 - 2(q_1^2 + q_2^2) \end{bmatrix} \quad (2.17)$$

## 제 3 장

### 기존의 위치측정시스템들과의 데이터 융합

이번 장에서는 시리얼 링크를 이용한 위치측정시스템을 사용하는 방법으로써 현재 널리 사용되고 있는 위치측정 시스템들과의 연동 방법에 대해서 서술한다. 기본적인 연동은 EKF 를 이용한 센서 융합이며 EKF 의 시스템 행렬을 얻는데 필요한 6-DOF 강체의 동역학 방정식을 서술하며 이를 바탕으로 실외환경에서의 위치측정시스템으로 보편화된 GPS/INS integration 시스템에 대해서 서술한다. GPS/INS integration 은 GPS 의 좌표계가 ECEF 또는 LLH 좌표계를 사용하고 속도벡터는 NED 좌표계를 사용한다. 그러나 실내환경이나 로컬환경에서는 속도벡터를 동체좌표계에서 측정된 데이터를 사용하므로 보다 더 간단하게 표현할 수 있다. 세 번째로 GPS/INS integration 방법을 간략화시킴으로써 본 논문에서 실험을 위해서 사용할 간략화된 시리얼 링크/INS integration 방법에 대해서 정리한다. 마지막으로 시리얼 링크를 위치 측정 시스템으로 사용하는데 있어서 Intelligent Lead 의 성능 지수를 정의하며 실내 환경과 실외 환경에서의 타 위치측정센서들과의 센서융합 방법에 대해서 제안한다.

### 3.1 6-DOF 강체 운동 방정식

6 DOF 강체의 일반화된 기구학 방정식은 다음과 같다.

$$\dot{P} = JV \quad (3.1)$$

위의 식에서  $P$  는 일반화된 위치로 관성좌표계에서 측정되는 위치 벡터  $p$  와 자세 벡터  $\rho$  로 이루어진다.

$$P = \begin{bmatrix} p^T & \rho^T \end{bmatrix}^T \quad (3.2)$$

위의 식에서  $V$  는 일반화된 속도 벡터로 동체좌표계에서 측정되는 병진속도 벡터  $v$  와 회전속도 벡터  $\omega$  로 이루어진다.

$$V = \begin{bmatrix} v^T & \omega^T \end{bmatrix}^T \quad (3.3)$$

$J$  는 관성좌표계와 동체좌표계사이의 속도 변환 행렬로써 다음과 같다.

$$J = \begin{bmatrix} C & 0 \\ 0 & R \end{bmatrix} \quad (3.4)$$

$C$  는 동체좌표계에서 측정된 병진 속도 벡터를 관성좌표계의 병진 속도 벡터로 변환하는 행렬로 앞 장의 식(2.1)의 전치행렬이 된다.

$$C_b^n = \begin{bmatrix} \cos \theta \cos \psi & (\sin \phi \sin \theta \cos \psi - \cos \phi \sin \psi) & (\cos \phi \sin \theta \cos \psi + \sin \phi \sin \psi) \\ \cos \theta \sin \psi & (\sin \phi \sin \theta \sin \psi + \cos \phi \cos \psi) & (\cos \phi \sin \theta \sin \psi - \sin \phi \cos \psi) \\ -\sin \theta & \sin \phi \cos \theta & \cos \phi \cos \theta \end{bmatrix} \quad (3.5)$$

R 은 동체좌표계에서 측정된 회전속도를 관성좌표계의 회전속도로 변환시켜주는 행렬로써 앞장의 식(2.2)와 동일하다.

$$R = \begin{bmatrix} 1 & \sin \phi \tan \theta & \cos \phi \tan \theta \\ 0 & \cos \phi & -\sin \phi \\ 0 & \frac{\sin \phi}{\cos \theta} & \frac{\cos \phi}{\cos \theta} \end{bmatrix} \quad (3.6)$$

6-DOF 강체의 일반화된 동역학 방정식은 다음과 같다.

$$M\dot{V} + NV = \Lambda \quad (3.7)$$

위의 수식에서 M 은 시스템관성행렬(system inertia matrix)로 다음과 같다.

$$M = \begin{bmatrix} mI_{3 \times 3} & 0 \\ 0 & I \end{bmatrix}, I = \begin{bmatrix} I_{xx} & 0 & 0 \\ 0 & I_{yy} & 0 \\ 0 & 0 & I_{zz} \end{bmatrix} \quad (3.8)$$

식 (3.8)에서  $I_{3 \times 3}$  은 단위행렬을 의미하며 아랫첨자 없는 I 는 관성행렬(inertial matrix)을 나타낸다.

N 은 구심력 행렬(centripetal matrix) 로 다음과 같다.

$$N = \begin{bmatrix} 0 & -mS(v) \\ 0 & -S(I\omega) \end{bmatrix} \quad (3.9)$$

위의 수식에서 S 는 왜대칭행렬(skew-symmetric matrix)를 의미한다.

$$S\left(\begin{bmatrix} a & b & c \end{bmatrix}^T\right) \equiv \begin{bmatrix} 0 & -c & b \\ c & 0 & -a \\ -b & a & 0 \end{bmatrix} \quad (3.10)$$

$\Lambda$  는 일반화된 외력 벡터 (Generalized reduced force vector)으로 다음과 같다.

$$\Lambda = \left[ \sum f \quad \sum \tau \right]^T \quad (3.11)$$

위의 기구학과 동역학 방정식을 위치와 자세로 나누어서 각각을 나타내면 다음과 같다.

$$\begin{aligned} \dot{p} &= Cv \\ \dot{\rho} &= R\omega \end{aligned} \quad (3.12)$$

$$\begin{aligned} m\dot{v} - mS(v)v &= \sum f \\ I\dot{\omega} - S(I\omega)\omega &= \sum \tau \end{aligned} \quad (3.13)$$

식 (3.13)에서 왜대칭행렬을 벡터 외적 형태로 나타내면 다음과 같다.

$$\begin{aligned} m\dot{v} + \omega \otimes (mv) &= \sum f \\ I\dot{\omega} + \omega \otimes (I\omega) &= \sum \tau \end{aligned} \quad (3.14)$$

식 (3.12)의 기구학방정식을 오일러 각 대신에 쿼터니언을 사용하면 다음과 같다.

$$\begin{aligned}\dot{p} &= Cv \\ \dot{q} &= \frac{1}{2}q \otimes \omega\end{aligned}\tag{3.15}$$

위치 벡터와 각도 벡터에 대한 동역학방정식을 구하면 다음과 같다. 먼저 식(3.15)에서 양변을 미분하면 다음과 같다.

$$\ddot{p} = C\dot{v} + \omega \otimes (Cv)\tag{3.16}$$

C가 회전변환 행렬이므로 다음의 식이 만족된다.

$$\omega \otimes (Cv) = C(\omega \otimes v)\tag{3.17}$$

식(3.14), 식(3.16)과 식(3.17)에 의해서

$$\ddot{p} = \frac{d}{dt}(Cv) = C(\dot{v} + \omega \otimes v) = C \frac{1}{m} \sum f\tag{3.18}$$

식(3.12)에서 회전운동의 기구학방정식 좌우 변을 한번 미분하고 식(3.14)를 대입하면

$$\ddot{p} = R\dot{\omega} = RI^{-1}\left(\sum \tau - \omega \otimes (I\omega)\right)\tag{3.19}$$

식(3.19)에서 동체좌표계에서 측정된 회전속도를 관성좌표계에서의 회전속도로 바꾸면 다음과 같다.

$$\ddot{p} = R\dot{\omega} = RI^{-1}\left(\sum \tau - (R^{-1}\dot{p}) \otimes (IR^{-1}\dot{p})\right)\tag{3.20}$$

위의 식(3.18)과 (3.20)은 6-DOF 강체의 동역학방정식으로 향후 Intelligent Lead의 끝 단에 부착된 IMU의 운동에 영향을 미치는



요소별로 파악해서 총 작용힘과 총 작용토크를 구한 후에 위의 두 식에 대입하면 IMU가 장착된 좌표계의 비선형 동역학 방정식이 구해진다.

### 3.2 GPS/INS Integration

GPS의 데이터는 위치 성분은 LLH 좌표계 또는 ECEF 좌표계를 기준으로, 속도 성분은 NED 좌표계를 기준으로 출력된다. 따라서 위의 세 좌표계에 대한 관계에 대한 이해를 필요로 한다. 아래의 그림은 ECEF 좌표계와 LLH 좌표계 그리고 NED 좌표계의 관계를 나타낸다.

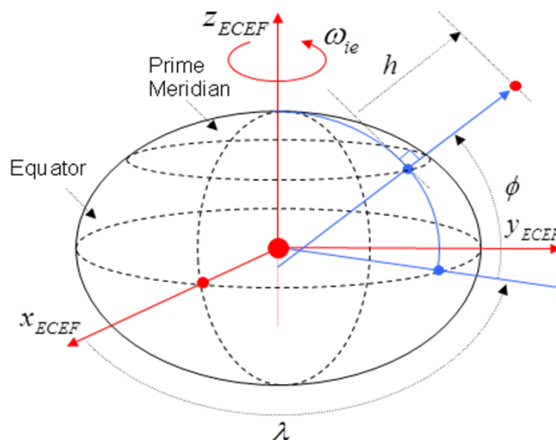


그림 3.1 ECEF 좌표계와 LLH 좌표계

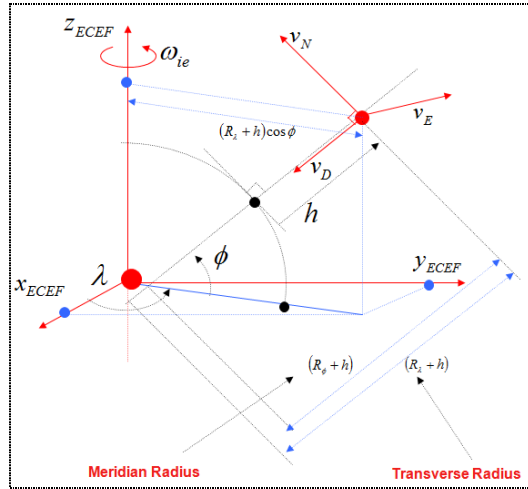


그림 3.2 LLH 좌표계와 NED 좌표계

먼저 GPS/INS Integration 시스템을 해석하는데 필요한 기호들을 나타내면 표 4.1 과 같다.

표 3.1 GPS/INS Integration 관련 기호 정의

Symbol	Meaning	Symbol	Meaning
$\phi$	위도	$\omega_{ie}$	지구중심좌표계의 회전 속도
$\lambda$	경도	$\omega_{en}$	동체좌표계의 회전속도
$h$	고도	$\omega_{in}$	항법좌표계의 회전속도
$v_N, v_E, v_D$	속도	$\gamma(\phi, h), \gamma(\phi)$	지역 중력 모델
$R_\phi$	Meridian Radius	$f_N, f_E, f_D$	NED 좌표계에서의 힘
$R_h$	Transverse Radius	$\zeta_g, \eta_g, \delta_g$	공기 저항력
$F$	시스템 행렬	$H$	출력 행렬

그림 4.1 의 아래쪽 그림으로부터 NED 좌표계로 표시되는 속도 벡터  $v = [v_N \ v_E \ v_D]^T$  와 LLH 좌표계로 표시되는 위치 벡터  $P = [\phi \ \lambda \ h]^T$  의 미분 관계식을 구하면 다음과 같다.

$$\begin{aligned}
v_N &= (R_\phi + h)\dot{\phi} \\
v_E &= (R_\lambda + h)\cos\phi\dot{\lambda} \\
v_D &= -\dot{h}
\end{aligned} \tag{3.21}$$

항법좌표계에서 측정된 관성좌표계를 기준으로 하는 지구의 회전속도를  $\omega_{ie}^n$  으로 정의하고, 항법좌표계에서 측정된 지구중심좌표계를 기준으로 하는 항법좌표계의 병진속도로 인해서 발생하는 회전속도를  $\omega_{en}^n$  으로 정의하면 위의 그림에 의해서 아래의 수식이 유도된다.

$$\begin{aligned}
\omega_{ie}^n &= \omega_{ie} [\cos\phi \quad 0 \quad -\sin\phi]^T \\
\omega_{en}^n &= \left[ \frac{v_E}{(R_\lambda + h)} \quad -\frac{v_N}{(R_\phi + h)} \quad -\frac{v_D \tan\phi}{(R_\lambda + h)} \right]^T
\end{aligned} \tag{3.22}$$

다음은 관성센서로 위치, 속도 및 자세를 측정하는 INS 시스템의 블록도를 나타낸다.

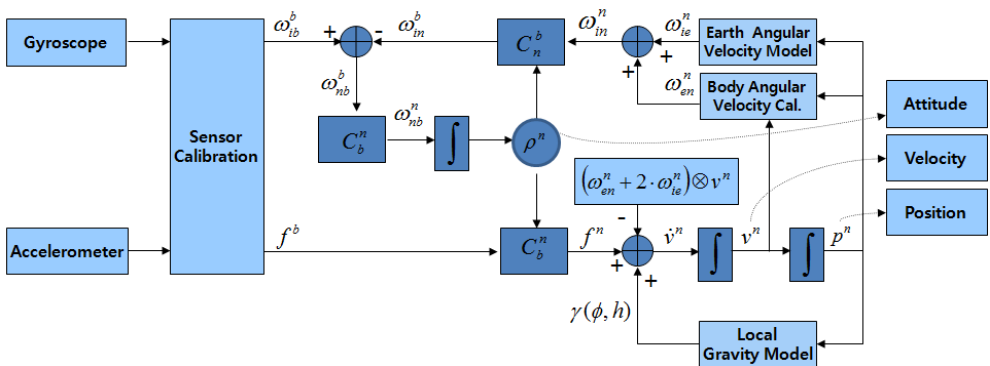


그림 3.3 INS 블록다이어그램

위의 그림에서 지구회전속도모델(earth angular velocity model)과 LLH 좌표계에서 측정된 동체회전속도(body angular velocity) 블록 관련 수식은 식(3.22)를 나타낸다.

위의 그림에서 지역중력모델(local gravity model)은 다음과 같다.

$$\begin{aligned} \gamma(\phi) &= \gamma_a \left[ 1 + (f_2 + f_4) \sin^2(\phi) - \frac{1}{4} f_4 \sin^2(2\phi) \right] \\ \gamma(\phi, h) &= \gamma(\phi) - \frac{2\gamma_a}{a} \left[ 1 + f + m + \left(-3f + \frac{5}{2}m\right) \sin^2(\phi) \right] h + \frac{3\gamma_a}{a^2} \end{aligned} \quad (3.23)$$

지역중력모델의 상수는 다음과 같이 구해진다.

$$a = 6378137.0(m)$$

$$b = 6356752.3(m)$$

$$\omega_{ie} = 7.2921151467 \times 10^{-5} (rad/s) = 15.0410671787 (deg/h)$$

$$\gamma_a = 9.780327 (m/s^2)$$

$$f = \frac{a-b}{a} = \frac{1}{298.257223563}$$

$$\frac{\omega_{ie}^2 a}{\gamma_a} = m + \frac{3}{2} m^2$$

$$f_2 = -f + \frac{5}{2} m + \frac{1}{2} f^2 - \frac{26}{7} f m + \frac{15}{4} m^2$$

$$f_4 = -\frac{1}{2} f^2 + \frac{5}{2} f m$$

수평축의 가속도계 바이어스(bias) 오차가 0.1mg 이면 약 0.1m rad.의 초기 자세오차로 나타나고 이로써 한 시간 후에 최대 약 0.7 해리(1 해리는 약 1,852 미터)의 위치오차가 발생하고, 수평축의

자이로스코프 드리프트(drift) 오차가 0.01 deg/h 이면 한 시간 후 약 0.7 해리의 위치오차가 발생하며, 수직축의 자이로스코프 드리프트(azimuth drift) 오차가 0.2 deg/h 이면 한 시간 후에 약 1 해리의 위치오차가 발생하는 것으로 알려져 있다. 이러한 INS 의 누적에러를 보상하기 위해서 GPS/INS integration 방법을 취한다.

식(3.21)을 EKF 의 시스템행렬을 구하기 위해서 행렬 형태로 나타내면 다음과 같다.

$$\begin{bmatrix} \dot{\phi} \\ \dot{\lambda} \\ \dot{h} \end{bmatrix} = \begin{bmatrix} \frac{1}{(R_\phi + h)} & 0 & 0 \\ 0 & \frac{1}{(R_\lambda + h)\cos\phi} & 0 \\ 0 & 0 & -1 \end{bmatrix} \begin{bmatrix} v_N \\ v_E \\ v_D \end{bmatrix} \quad (3.24)$$

시스템행렬, F 의 일부를 형성하는 위치에 대한 미분 값은 다음과 같다.

$$F_{rr} = \begin{pmatrix} \frac{\partial \dot{\phi}}{\partial \phi} & \frac{\partial \dot{\phi}}{\partial \lambda} & \frac{\partial \dot{\phi}}{\partial h} \\ \frac{\partial \dot{\lambda}}{\partial \phi} & \frac{\partial \dot{\lambda}}{\partial \lambda} & \frac{\partial \dot{\lambda}}{\partial h} \\ \frac{\partial \dot{h}}{\partial \phi} & \frac{\partial \dot{h}}{\partial \lambda} & \frac{\partial \dot{h}}{\partial h} \end{pmatrix} = \begin{pmatrix} 0 & 0 & \frac{-v_N}{(R_\phi + h)^2} \\ \frac{v_E \sin\phi}{(R_\lambda + h)\cos^2\phi} & 0 & \frac{-v_E}{(R_\lambda + h)^2 \cos\phi} \\ 0 & 0 & 0 \end{pmatrix} \quad (3.25)$$

$$F_{rv} = \begin{pmatrix} \frac{\partial \dot{\phi}}{\partial v_N} & \frac{\partial \dot{\phi}}{\partial v_E} & \frac{\partial \dot{\phi}}{\partial v_D} \\ \frac{\partial \dot{\lambda}}{\partial v_N} & \frac{\partial \dot{\lambda}}{\partial v_E} & \frac{\partial \dot{\lambda}}{\partial v_D} \\ \frac{\partial \dot{h}}{\partial v_N} & \frac{\partial \dot{h}}{\partial v_E} & \frac{\partial \dot{h}}{\partial v_D} \end{pmatrix} = \begin{pmatrix} \frac{1}{R_\phi + h} & 0 & 0 \\ 0 & \frac{1}{(R_\lambda + h)\cos\phi} & 0 \\ 0 & 0 & -1 \end{pmatrix}$$

3.1 절에서 구한 동역학 방정식 식 (3.13)에서 위치에 관한 동역학 방정식에서 외부 힘에 중력과 코리올리 힘을 추가 하여 정리하면 다음과 같다.

$$\frac{d}{dt}v^n = f^n - (\omega_{en}^n + 2\omega_{ie}^n) \times v^n + g^n \quad (3.26)$$

식 (3.22)로부터  $(\omega_{en}^n + 2\omega_{ie}^n) \times v^n$  을 구하면 다음과 같다.

$$(\omega_{en}^n + 2\omega_{ie}^n) \times v^n = \begin{bmatrix} -v_E \left[ \frac{v_E}{(R_\lambda + h) \cos \phi} + 2\omega_{ie} \right] \sin \phi + \frac{v_N v_D}{(R_\phi + h)} \\ v_N \left[ \frac{v_E}{(R_\lambda + h) \cos \phi} + 2\omega_{ie} \right] \sin \phi + v_D \left[ \frac{v_E}{(R_\lambda + h) \cos \phi} + 2\omega_{ie} \right] \cos \phi \\ -v_E \left[ \frac{v_E}{(R_\lambda + h) \cos \phi} + 2\omega_{ie} \right] \cos \phi - \frac{v_N^2}{(R_\phi + h)} \end{bmatrix} \quad (3.27)$$

최종적인 속도 방정식은 다음과 같다.

$$\begin{bmatrix} \dot{v}_N \\ \dot{v}_E \\ \dot{v}_D \end{bmatrix} = \begin{bmatrix} f_N \\ f_E \\ f_D \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ \gamma(\phi, h) \end{bmatrix} + \begin{bmatrix} \zeta_g \\ \eta_g \\ \delta_g \end{bmatrix} + \begin{bmatrix} -v_E \left[ \frac{v_E}{(R_\lambda + h) \cos \phi} + 2\omega_{ie} \right] \sin \phi + \frac{v_N v_D}{(R_\phi + h)} \\ v_N \left[ \frac{v_E}{(R_\lambda + h) \cos \phi} + 2\omega_{ie} \right] \sin \phi + v_D \left[ \frac{v_E}{(R_\lambda + h) \cos \phi} + 2\omega_{ie} \right] \cos \phi \\ -v_E \left[ \frac{v_E}{(R_\lambda + h) \cos \phi} + 2\omega_{ie} \right] \cos \phi - \frac{v_N^2}{(R_\phi + h)} \end{bmatrix} \quad (3.28)$$

식 (3.28)로부터 시스템행렬의 일부를 형성하는 위치에 대한 미분 값은 다음과 같이 유도된다.

$$\begin{aligned}
F_{v^*} &= \begin{pmatrix} \frac{\partial \dot{v}_N}{\partial \phi} & \frac{\partial \dot{v}_N}{\partial \lambda} & \frac{\partial \dot{v}_N}{\partial h} \\ \frac{\partial \dot{v}_E}{\partial \phi} & \frac{\partial \dot{v}_E}{\partial \lambda} & \frac{\partial \dot{v}_E}{\partial h} \\ \frac{\partial \dot{v}_D}{\partial \phi} & \frac{\partial \dot{v}_D}{\partial \lambda} & \frac{\partial \dot{v}_D}{\partial h} \end{pmatrix} = \begin{pmatrix} -2v_E \omega_{ie} \cos \phi - \frac{v_E^2}{(R_\lambda + h) \cos^2 \phi} & 0 & \frac{-v_N v_D}{(R_\phi + h)^2} + \frac{v_E^2 \tan \phi}{(R_\lambda + h)^2} \\ 2\omega_{ie} (v_N \cos \phi - v_D \sin \phi) + \frac{v_E v_N}{(R_\lambda + h) \cos^2 \phi} & 0 & \frac{-v_E v_D}{(R_\lambda + h)^2} - \frac{v_N v_E \tan \phi}{(R_\lambda + h)^2} \\ 2v_E \omega_{ie} \sin \phi & 0 & \frac{v_E^2}{(R_\lambda + h)^2} + \frac{v_N^2}{(R_\phi + h)^2} - 2\frac{\gamma_a}{a} \end{pmatrix} \quad (3.29) \\
F_{v^*} &= \begin{pmatrix} \frac{\partial \dot{v}_N}{\partial v_N} & \frac{\partial \dot{v}_N}{\partial v_E} & \frac{\partial \dot{v}_N}{\partial v_D} \\ \frac{\partial \dot{v}_E}{\partial v_N} & \frac{\partial \dot{v}_E}{\partial v_E} & \frac{\partial \dot{v}_E}{\partial v_D} \\ \frac{\partial \dot{v}_D}{\partial v_N} & \frac{\partial \dot{v}_D}{\partial v_E} & \frac{\partial \dot{v}_D}{\partial v_D} \end{pmatrix} = \begin{pmatrix} \frac{v_D}{R_\phi + h} & -2\omega_{ie} \sin \phi - 2\frac{v_E \tan \phi}{R_\lambda + h} & \frac{v_N}{R_\phi + h} \\ 2\omega_{ie} \sin \phi + \frac{v_E \tan \phi}{R_\lambda + h} & \frac{v_D + v_N \tan \phi}{R_\lambda + h} & 2\omega_{ie} \cos \phi + \frac{v_E}{R_\lambda + h} \\ -2\frac{v_N}{R_\phi + h} & -2\omega_{ie} \cos \phi - 2\frac{v_E}{R_\lambda + h} & 0 \end{pmatrix}
\end{aligned}$$

3.1 절에서 구한 기구학 방정식, 식 (3.12) 에서 자세에 관한 기구학 방정식을 정리하면 다음과 같다.

$$\frac{d}{dt} \rho^n = C_b^n \omega_{nb}^b = C_b^n (\omega_{ib}^b - \omega_m^b) = C_b^n (\omega_{ib}^b - C_n^b \omega_m^n) = \omega_{ib}^n - (\omega_{ie}^n + \omega_{en}^n) \quad (3.30)$$

식(3.30)의  $\omega_{ie}^n$  과  $\omega_{en}^n$  은 (3.22)에서 정의되어 있으며, 위의 수식으로부터 시스템행렬의 일부를 형성하는 자세에 대한 미분 값은 다음과 같이 유도된다.

$$\begin{aligned}
F_{\rho^*} &= \begin{pmatrix} \frac{\partial \dot{\phi}_e}{\partial \phi} & \frac{\partial \dot{\phi}_e}{\partial \lambda} & \frac{\partial \dot{\phi}_e}{\partial h} \\ \frac{\partial \dot{\theta}_e}{\partial \phi} & \frac{\partial \dot{\theta}_e}{\partial \lambda} & \frac{\partial \dot{\theta}_e}{\partial h} \\ \frac{\partial \dot{\psi}_e}{\partial \phi} & \frac{\partial \dot{\psi}_e}{\partial \lambda} & \frac{\partial \dot{\psi}_e}{\partial h} \end{pmatrix} = \begin{pmatrix} \omega_{ie} \sin \phi & 0 & \frac{v_E}{(R_\phi + h)^2} \\ 0 & 0 & \frac{-v_N}{(R_\lambda + h)^2} \\ \omega_{ie} \cos \phi + \frac{v_E}{(R_\lambda + h) \cos^2 \phi} & 0 & \frac{-v_E \tan \phi}{(R_\lambda + h)^2} \end{pmatrix} \quad (3.31) \\
F_{\rho^*} &= \begin{pmatrix} \frac{\partial \dot{\phi}_e}{\partial v_N} & \frac{\partial \dot{\phi}_e}{\partial v_E} & \frac{\partial \dot{\phi}_e}{\partial v_D} \\ \frac{\partial \dot{\theta}_e}{\partial v_N} & \frac{\partial \dot{\theta}_e}{\partial v_E} & \frac{\partial \dot{\theta}_e}{\partial v_D} \\ \frac{\partial \dot{\psi}_e}{\partial v_N} & \frac{\partial \dot{\psi}_e}{\partial v_E} & \frac{\partial \dot{\psi}_e}{\partial v_D} \end{pmatrix} = \begin{pmatrix} 0 & \frac{-1}{(R_\lambda + h)} & 0 \\ \frac{1}{R_\phi + h} & 0 & 0 \\ 0 & \frac{\tan \phi}{(R_\lambda + h)} & 0 \end{pmatrix}
\end{aligned}$$

식(3.25), 식(3.29)와 식(3.31)을 이용해서 시스템행렬을 다음과 같이 업데이트시킨다.

$$F = \begin{pmatrix} F_{rr} & F_{rv} & 0 \\ F_{vr} & F_{vv} & -S(f^n) \\ F_{\rho r} & F_{\rho v} & -S(\omega_{ib}^n) \end{pmatrix} \quad (3.32)$$

EKF 의 측정항 갱신(measurement update)을 위해서 GPS 로부터 LLH 좌표 및 NED 속도를 획득한 후에 INS 의 위치 및 속도 데이터와 다음과 같이 융합한다.

$$H_k x_k = \begin{pmatrix} (R_\phi + h)(\phi_{INS} - \phi_{GPS}) \\ (R_\lambda + h)\cos\phi(\lambda_{INS} - \lambda_{GPS}) \\ h_{INS} - h_{GPS} \\ \text{-----} \\ v_{INS}^n - v_{GPS}^n \end{pmatrix} \quad (3.33)$$

식 (3.33)에서 아랫 첨자 INS와 GPS는 각각 INS와 GPS에서 측정된 위치 및 속도 데이터를 의미한다. 앞의 과정에 의해서 계산되는 GPS/INS Integration 기법의 단점은 정밀도가 GPS에 너무 의존적이라는 것이다. 또한 GPS는 벽면 근처에서 수신 감도가 급격히 떨어져서 벽면 근처에서 작업을 하는 구조용 로봇에 사용하기 어려우며 고의 신호로 GPS를 교란하는 것도 큰 문제가 된다. 본 논문에서 제안하는 Intelligent Lead에서 생성되는 위치데이터와 GPS/INS integration 에 의해서 생성되는 위치 데이터를 합성하기 위해서는 GPS/INS integration 의 성능지수를 정하는 것이 필요하다. 위성들의 상대적인 기하학이 위치 결정에 미치는 오차를 나타내는 무 차원의 수인 DOP(Dilution of



Precision)가 GPS의 성능지수로 사용하는 것이 일반적이며, GPS의 DOP가 임계 값 이상으로 올라가면 GPS 데이터를 EKF의 어떤 측정값 갱신 단계에서 빼는 것이 간단한 융합 방법이다.

### 3.3 시리얼 링크/INS Integration

위에서 서술한 GPS/INS integration 은 GPS 의 좌표계가 ECEF 또는 LLH 좌표계를 사용하고 속도벡터는 NED 좌표계를 사용한다. 그러나 실내환경이나 로컬환경에서는 속도벡터를 동체좌표계에서 측정된 데이터를 사용하므로 보다 더 간단하게 표현할 수 있다. 또한 코리올리 힘을 무시하고 중력가속도를 상수로 둬으로써 수식이 간단해 진다.

3.1 장에서 구한 기구학 방정식과 동역학 방정식을 항법좌표계에서의 측정값을 첨자  $n$  으로 동체좌표계에서의 측정값을 첨자  $b$  로 표기하면 다음과 같다.

$$\dot{p}_n = C_b^n v_b \quad (3.34)$$

$$m\dot{v}_b + \omega_b \otimes (mv_b) = \sum f \quad (3.35)$$

$$\dot{q} = \frac{1}{2} q \otimes \omega_b \quad (3.36)$$

외력의 합은 중력에 의한 힘과 가속도 센서에 의해서 측정되는 힘으로 나뉜다.

$$\sum f = f_b + mC_n^b g_n \quad (3.37)$$

식(3.35)를 위의 수식을 대입하여 정리하면 다음과 같다.

$$\dot{v}_b = \frac{1}{m} f_b + C_n^b g_n - \omega \otimes v_b \quad (3.38)$$

벡터 외적의 순서를 바꾸어 식(3.36)를 표시하면 다음과 같다.

$$\dot{q} = \frac{1}{2} q \otimes \omega = -\frac{1}{2} \omega \otimes q \quad (3.39)$$

위의 수식(3.37), (3.38), (3.39)은 비선형 방정식으로 EKF 에서는 타입 스텝마다 선형화에 의해서 A, H 행렬을 구하고 EKF 에 대입하여 P 행렬과 상태벡터를 업데이트시킨다. 각속도는 자이로스코프에 의해서 측정되므로 추정하는 요소를 위치와 속도와 쿼터니언으로 두어서 상태벡터를 만들고 출력벡터를 위치, 속도, 자세로 두면 다음과 같다.

$$x = \begin{bmatrix} p_n \\ v_b \\ q \end{bmatrix}, y = \begin{bmatrix} p_n \\ v_b \\ \theta \end{bmatrix} \quad (3.40)$$

EKF 의 시스템행렬, A 와 출력행렬, H 는 비선형 방정식을 상태변수로 선형화하여 구해지며 정의는 다음과 같다.

$$A \equiv \frac{\partial f(x, u)}{\partial x}, H \equiv \frac{\partial h(x, u)}{\partial x} \quad (3.41)$$

위의 정의에 의해서 A, H 행렬을 나타내면 다음과 같다.

$$A = \begin{bmatrix} 0 & (C_n^b)^T & \frac{\partial(C_n^b v_b)}{\partial q} \\ 0 & -\omega \otimes & \frac{\partial(C_n^b g_n)}{\partial q} \\ 0 & 0 & -\frac{1}{2}\omega \otimes \end{bmatrix} \quad (3.42)$$

$$H = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & \frac{\partial \theta}{\partial q} \end{bmatrix} \quad (3.43)$$

왜대칭 행렬로 표현하면 다음과 같다.

$$A = \begin{bmatrix} 0 & (C_n^b)^T & \frac{\partial(C_n^b v_b)}{\partial q} \\ 0 & -S(\omega) & \frac{\partial(C_n^b g_n)}{\partial q} \\ 0 & 0 & -\frac{1}{2}S(\omega) \end{bmatrix} \quad (3.44)$$

수식 (3.42)과 (3.44)에서 사용되는  $C_n^b$  는 식 (3.5)의 전치행렬을 나타내며 정규화(normalize) 된 쿼터니언으로 표시하면 다음과 같다.

$$C_n^b = \begin{bmatrix} 1-2(q_2^2 + q_3^2) & 2(q_0q_3 + q_1q_2) & 2(-q_0q_2 + q_1q_3) \\ 2(-q_0q_3 + q_1q_2) & 1-2(q_1^2 + q_3^2) & 2(q_0q_1 + q_2q_3) \\ 2(q_0q_2 + q_1q_3) & 2(-q_0q_1 + q_2q_3) & 1-2(q_1^2 + q_2^2) \end{bmatrix} \quad (3.45)$$

위의 수식들에서 사용된 편도함수(partial derivative)항목들은 식 3.45 를 쿼터니언 원소들로 편미분 함으로써 구해진다.

최종 구해진 시스템행렬과 출력행렬은 EKF 의 갱신단계에 쓰이게 된다. 칼만 필터의 시간예측갱신단계(time prediction update procedure)는 손목에 장착된 IMU 에 의해서 구현한다. 측정값 갱신(measurement update)은 시리얼링크의 순기구학을 풀어서 구해진 손목의 위치 데이터로 업데이트 한 후에 LIDAR 센서에 의해서 획득되는 어깨 위치와 손목의 위치로부터 추정되는 팔꿈치의 각을 업데이트 한다. 이전 단계의 팔꿈치의 각과 LIDAR 센서에 의해서 획득되는 어깨 위치로 손목의 위치를 추정하며 위의 데이터를 측정값 갱신 시에 이용한다.

EKF 수행절차에서 시간 예측 갱신 단계를 관련된 수식으로 세부 서술하면 다음과 같다.

- 1) 현재의 상태벡터와 IMU 센서 데이터를 이용해서 상태벡터를 업데이트한다.  $x_n = x_{n-1} + \dot{x}_{n-1}dt$
- 2) 업데이트된 상태값들 중에서 쿼터니언을 정규화시킨다.
- 3) 현재의 상태벡터로부터 시스템행렬 A 를 계산한다.
- 4) P 행렬을 업데이트한다.  $P_n = AP_{n-1}A^T + Q$

또한 측정값 갱신 단계를 관련된 수식으로 세부 서술하면 다음과 같다.

- 1) 현재의 쿼터니언 값으로부터 출력행렬 H 를 구한다.
- 2) 출력값 에러를 계산한다.  $e = z_m - Hx_n$
- 3) 칼만 게인(gain) 을 업데이트한다.  $K = P_n H^T [HP_n H^T + R]^{-1}$

4) 상태벡터를 업데이트한다.  $x_{n+1} = x_n + Ke$

5) P 행렬을 업데이트한다.  $P_{n+1} = P_n - KHP_n$

위의 과정에서 업데이트되는 상태변수의 첫 3항은 Intelligent Lead의 출력값을 연산을 위해서 필요로 하는 기준좌표계에서 바라본 손목의 위치 좌표가 된다.

### 3.4 Intelligent Lead의 성능 지수

시리얼 링크의 경우에 기준좌표계가 지면이나 벽면 또는 로봇에 부착되지 않은 경우에는 위치 및 자세 정보를 획득할 수 없게 된다. 또한 돌발상황에 의해서 부착된 것이 떨어지는 경우도 발생할 수 있다. 따라서 부착 유무를 확인하기 위해서 기준좌표계에 힘센서(force sensor)를 달고 측정되는 값이 임계 값 이하가 되면 EKF의 측정치 항에서 제외시키는 경우를 생각할 수 있다. 시리얼 링크의 기준좌표계와 손목 사이의 거리가 어느 정도 확보되어야만 핸들을 자유롭게 움직이는 것이 가능하다. 극단적인 예로 시리얼 링크의 모든 링크들이 일직선상에 있으면 최대로 벌어진 상태로써 더 이상 z방향의 위치 변화를 측정할 수 없게 된다. 따라서 위의 방법 만으로는 안정적인 위치데이터를 획득하기 어렵다. 본 논문에서는 직렬형 로봇의 특이점 회피 방법으로 Nakamura에 의해서 정의된 MOM(Measure of Manipulability)를 사용함으로써 위의 문제를 해결한다.

직렬형 로봇의 MOM 은 다음과 같이 구해진다.

$$MOM = \sqrt{\det(J \cdot J^T)} \quad (3.46)$$

본 논문에서 실험용으로 제작한 Intelligent Lead 는 구형(spherical) 관절을 가진 구조로써 구형 관절은 3 축이 한점에 모여있는 구조로 해석하면 3 개의 구형 관절로 이루어진 Intelligent Lead 는 여유자유도(redundancy)를 가진 9 축 로봇의 자코비안을 구하는 과정과 동일하다. 링크 끝 단의 한쪽은 로봇에 부착되며 다른 한쪽은 사용자가 손으로 잡게 된다.

MOM 의 증가는 Intelligent Lead 의 정확도가 감소함을 의미한다. 따라서 다른 위치측정 센서와 융합할 경우에 MOM 을 가중치요소(weighting factor)로 두거나 MOM 이 특정 값이 상으로 올라가면 EKF 의 측정값 업데이트에서 제외시키면 된다. 다른 위치측정센서로 실외용의 GPS 의 경우에는 DOP 로 성능지수를 잡을 수 있다. 비정형 환경에서 실내용 위치 측정 수단으로 고려할 수 있는 것은 Visual SLAM 에 의해서 이동량 측정 방법이다. Visual SLAM 에 의한 위치 및 자세 값을 구하는 과정은 다음과 같다. 먼저 RGB 데이터로부터 KLT feature tracker 를 이용하여 특이점(feature) 추출(extraction) 및 특이점 정합(matching)을 수행한다. 정합된 특이점 쌍들로부터 RANSAC 알고리즘과 ICP 알고리즘을 이용하여 이전 프레임(frame)에 대한 현재 프레임의 병진운동의 이동 량과 회전운동의 이동 량을 산출한다. Visual

SLAM 에서의 성능 저하는 크게 두 가지 원인에 의해서 일어난다. 먼저 특이점 추출 시의 성능저하는 연무가 일어나거나 환경이 너무 단순한 경우에 발생한다. 두 번째로 특이점 정합 시의 성능 저하는 주변환경의 변화 즉 동적 요소가 어느 정도 인가에 따라서 좌우된다. 따라서 위의 두 요소를 고려해서 특이점 추출 수와 특이점 정합 수 및 정합 에러를 이용해서 성능 지수를 나타내는 방법을 생각할 수 있다. OpenCV 를 이용해서 특이점을 추출 및 정합하는 경우에 특이점 추출 수는 GoodFeatureToTrack 함수를 이용할 때 추출된 특이점 수로 하고 정합 수는 KLT 알고리즘에 의해서 정합된 수를 취하는 것이 하나의 예가 된다.

이렇게 설정한 성능 지수가 임계 값 이하가 되면 EKF 의 측정치  
항 갱신 단계에서 추가되는 것을 막거나 측정값들을 융합할 때  
가중치로 사용하는 것을 고려할 수 있다.

## 제 4 장

### Intelligent Lead를 이용한 지능형 안내 시스템

본 장에서 2장에서 기술한 Intelligent Lead와 이동 로봇에 부착된 LIDAR를 이용해서 1장의 2절에서 서술한 기능을 가지는 안내 로봇을 제작하는 과정을 서술한다. 먼저 LIDAR로부터 측정되는 사용자의 몸체 및 어깨의 위치와 시리얼 링크로부터 측정되는 사용자의 손목의 위치를 이용해서 Intelligent Lead의 출력값을 생성하는 알고리즘에 관해서 기술한다.

두 번째로 GuideCane에서 사용된 VFH+알고리즘을 변경시켜서 Intelligent Lead의 출력값을 입력으로 받아들여서 로봇을 제어할 수 있는 모션 제어 알고리즘에 대해서 기술한다. 위의 제어기는 사용자가 Intelligent Lead를 통해서 로봇에게 유입되는 회전 및 속도 제어 명령을 로봇의 모션제어기에서 추종할 수 있으며 또한 장애물 탐지 후에 장애물 회피 알고리즘이 구동 될 수 있어야 한다. 또한 사용자의 위치를 파악해서 미리 입력된 사용자와의 간격을 유지하는 기능을 가지고 있어야 한다.

마지막으로 제작된 이동 로봇과 사용된 센서들 및 제작된 Intelligent Lead에 대해서 살펴보며 각각의 부분들이 안내 로봇의 주요 기능 구현을 위해서 어떻게 쓰이는 지에 대해서 설명한다.



## 4.1 Intelligent Lead의 출력값 생성 알고리즘

3장에서 구한 EKF의 출력으로 사용자의 손목의 위치가 구해지면 LIDAR로부터 획득되는 사용자의 오른쪽 어깨, 왼쪽 어깨, 몸체의 위치를 이용하여 사용자의 제어 의도를 파악하는 과정을 거쳐야만 한다. 아래의 표는 Intelligent Lead의 출력량을 수식으로 나타내는 과정에서 필요로 하는 기호들을 나타낸다.

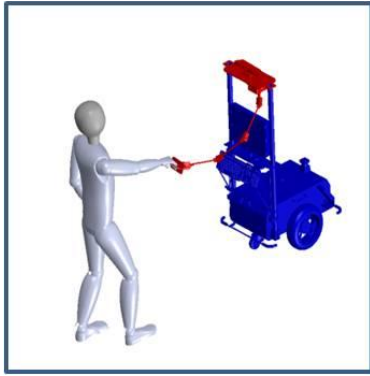
표 4.1 Intelligent Lead 출력 관련 기호 정의

Symbol	Meaning	Symbol	Meaning
$p_H$	손의 위치	$O_S$	속도 제어 량
$p_{SL}$	왼쪽 어깨의 위치	$O_T$	회전 제어 량
$p_{SR}$	오른쪽 어깨의 위치	$O_D$	로봇과 사용자와의 거리
$P_T$	몸체의 위치	$D_{SH}$	손목과 어깨와의 거리
$\theta_E$	팔꿈치 각	$\vec{n}$	법선 벡터
$\theta_V$	수직 각	$L_{UA}$	윗 팔 길이
$\theta_H$	수평 각	$L_{LA}$	아래 팔 길이

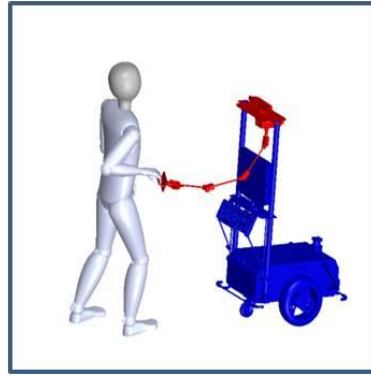
사용자마다 보행의 보폭과 팔의 앞뒤 이동 및 좌우 이동의 크기가 다르기 때문에 보행 시의 팔 관절 변화에 대한 정보가 필요하다. 사용자 관절의 위치 변화와 사용자의 제어 의도간의 관계를 깊이 있게 파악하는 것은 본 논문의 영역 밖이며 본 논문에서는 실험을 위해서 직관적 방법으로 간단하게 제어 규칙을 정했다.

강아지를 끌고 산책할 경우를 생각해보면 보행 시에 에너지 사용을 최소화하는 방향으로 손목의 위치가 변화된다. 손목을

좌우로 각도를 꺾어서 방향을 지시하기 보다는 손목의 위치를 좌우로 보내는 것이 일반적이다. 또한 속력의 증감은 손목을 앞으로 밀거나 뒤로 끌어당기는 것이 일반적이다. 아래의 그림은 사용자가 속도 증가, 속도 감소, 왼쪽 회전, 오른쪽 회전 명령을 내릴 때의 상황을 묘사한 것이다.



(a) 속도 증가



(b) 속도 감소



(c) 왼쪽 회전



(d) 오른쪽 회전

그림 4.1 사용자의 제어 모션

사람마다 걷는 보폭이 다르고 앞뒤로 팔을 흔드는 양 또한 다르다. 따라서 걷는 동안에 사용자의 팔꿈치가 이루는 각을 측정할

필요가 있다. 본 논문에서는 사용자가 오른 손잡이로 가정한다. 아래의 그림은 Intelligent Lead의 출력을 결정하기 위해서 정의되는 좌표계, 각과 거리에 대한 정의를 보여준다.

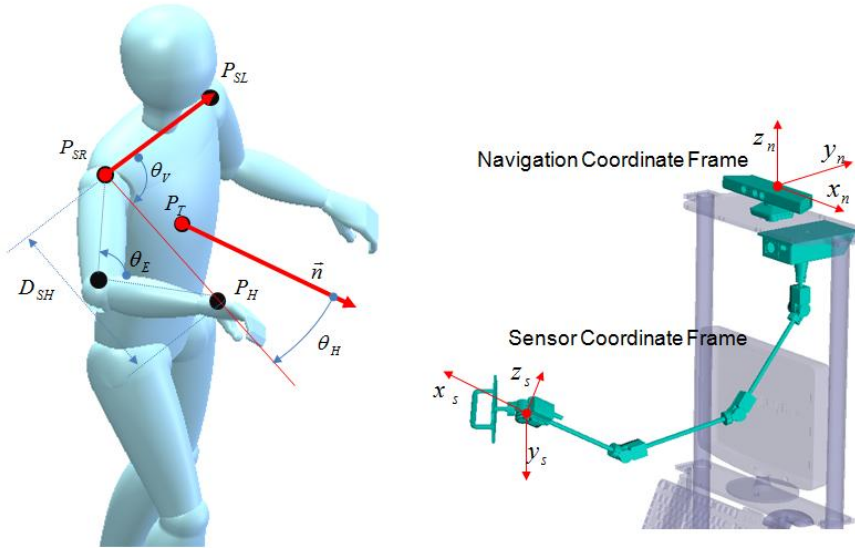


그림 4.2 좌표계, 각과 거리 정의

위의 그림에서 법선 벡터,  $n$  은 사용자의 왼쪽어깨와 오른쪽 어깨 및 몸체 중심의 좌표점이 생성하는 평면의 방향 벡터를 의미한다. 아래의 그림은 법선 벡터의 의미를 명확히 하기 위해서 사용자만을 그린 것이다.

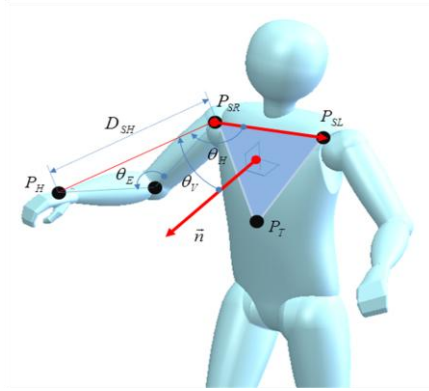


그림 4.3 법선 벡터 정의

$D_{SH}$  은 오른쪽 어깨와 손목의 위치가 만드는 선분의 크기를 의미한다.  $\theta_V$  는 법선 벡터와 오른쪽 어깨점과 손목이 만드는 선분과의 각을 나타낸다.  $\theta_H$  는 오른쪽 어깨점과 손목이 만드는 선분과 두 어깨점이 만드는 선분이 이루는 각으로 정의된다.

사용자의 회전 제어 량과 속도 제어 량은 몸체와 어깨로부터 손목으로 그은 선분이 이루는 각을 이용해서 구해진다.

먼저 사용자와 로봇간의 거리는 다음과 같이 구해진다.

$$D_{SH} = \|P_H - P_{SR}\| \quad (4.1)$$

그림 4.3 에서 정의된 각각의 각도들을 코사인법칙을 사용해서 구하면 다음과 같다.

$$\theta_E = \pi \pm \arccos\left(\frac{L_{UA}^2 + L_{LA}^2 - D_{SH}^2}{2L_{UA} \cdot L_{LA}}\right) \quad (4.2)$$

$$\theta_V = \arccos\left(\frac{\vec{n} \cdot (P_H - P_{SR})}{\|\vec{n}\| \times \|P_H - P_{SR}\|}\right) \quad (4.3)$$

$$\theta_H = \arccos\left(\frac{(P_{SL} - P_{SR}) \cdot (P_H - P_{SR})}{\|P_{SL} - P_{SR}\| \times \|P_H - P_{SR}\|}\right) \quad (4.4)$$

식 (4.3)에서 법선 벡터는 왼쪽 어깨, 오른쪽 어깨, 몸체의 위치를 포함하는 평면의 법선 벡터이며 다음과 같이 구해진다.

$p_H = (x_1 \ y_1 \ z_1)^T$ ,  $p_{SL} = (x_2 \ y_2 \ z_2)^T$ ,  $p_{SR} = (x_3 \ y_3 \ z_3)^T$  인 경우에

$$\vec{n} = \left( \left( \begin{array}{cc|cc|cc} y_2 - y_1 & z_2 - z_1 & x_2 - x_1 & z_2 - z_1 & x_2 - x_1 & y_2 - y_1 \\ y_3 - y_1 & z_3 - z_1 & x_3 - x_1 & z_3 - z_1 & x_3 - x_1 & y_3 - y_1 \end{array} \right)^T \right) \quad (4.5)$$

만약 손목의 높이가 몸체 중심점 보다 위에 있다면 사용자가 로봇을 제어할 의도를 가지고 있는 것으로 가정하고 Intelligent Lead의 출력 값을 출력하며 그렇지 않은 경우에는 0으로 설정한다. 또한  $\theta_V$ 가 임계 값 보다 크다면 사용자가 허리를 굽히는 것으로 가정한다. 위의 경우에도 출력 값은 0으로 설정한다. 실험 시에는 임계 값을 45도로 설정하고 테스트를 했다. 이러한 가정은 의도되지 않은 출력을 발생시킴으로써 로봇의 원치 않는 행동을 하는 것을 막을 수 있다. 트림버튼을 누르면 속도제어 및 회전제어를 위한 트림 값들을 설정한다.

아래의 값은 기본값(default value)으로 설정된 값이다.

$$TRIM_{SPEED} = \sqrt{L_{UA}^2 + L_{LA}^2}, TRIM_{TURN} = \frac{\pi}{2} \quad (4.6)$$

최종적으로 Intelligent Lead의 출력은 다음과 같다.

$$O_S = \begin{cases} D_{SH} - TRIM_{SPEED} & , \text{if } \theta_V < \frac{\pi}{4} \text{ and } P_H(z) > P_T(z) \\ 0 & , \text{otherwise} \end{cases} \quad (4.7)$$

$$O_T = \begin{cases} \theta_H - TRIM_{TURN} & , \text{if } \theta_V < \frac{\pi}{4} \text{ and } P_H(z) > P_T(z) \\ 0 & , \text{otherwise} \end{cases} \quad (4.8)$$

위의 수식 (4.7)과 (4.8) 에서  $O_S$  와  $O_T$  는 각각 속도제어량과 회전제어량을 나타낸다. 또한  $P_H(z)$  와  $P_T(z)$  는 손목과 몸체 중심의  $z$ 축 값이 된다. 로봇과 사용자와의 거리는 간단하게 두 개의 어깨점과 몸체중심점의 평균 값을 구하는 것으로 설정한다.

$$O_D = \frac{\|P_{SL} + P_{SR} + P_T\|}{3} \quad (4.9)$$

아래의 그림.4.4는 안내 로봇에 적용되는 Intelligent Lead의 블록다이어그램이다.

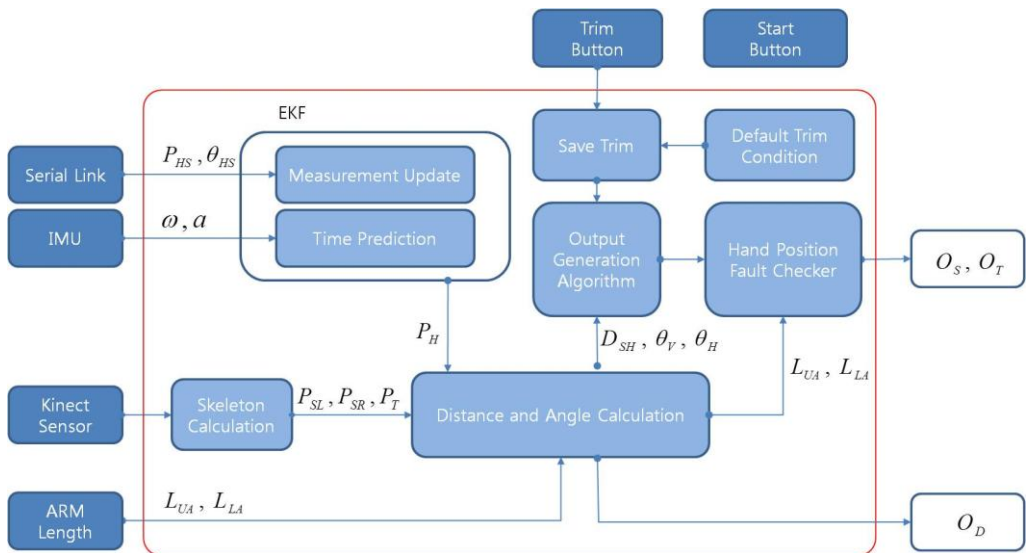


그림 4.4 Intelligent Lead의 블록다이어그램.

Intelligent Lead는 수식(4.7), (4.8) 그리고 (4.9)에서 나타나는  $O_D, O_S, O_T$  를 실시간으로 동시에 출력하게 된다.

## 4.2 이동 로봇의 모션 제어 알고리즘

미지의 환경(unknown environment)에서 장애물과의 충돌 회피나 로봇 간의 충돌 회피를 구현하기 위한 많은 선행연구들이 이루어졌다. PFM(potential field method), dynamic window (DW), velocity obstacle(VO), vector field histogram(VFH) 그리고 vector polar histogram(VPH) 를 사용하는 방법 등이 연구되었으며 이중에서도 VFH[8,9]와 VPH는 광범위하고도 유용한 결과를 가져왔다.

가상 척력(repulsive force)과 가상 인력(attractive force)을 사용하여 로봇제어 신호를 생성하는 PFM[26]은 간단하고 구현이 쉬워서 흔히 사용되었으나 국소최저치(local minima)에서 움직일 수 없는 상태에 빠지거나 많은 장애물 들을 가진 복잡한 환경에서 길을 찾지 못하거나, 좁은 복도에서 진동(oscillation) [27]하는 단점을 가지고 있다. VO를 사용하는 방법은 로봇과 장애물들 사이에 충돌을 야기시키는 속도 집합을 결정하는 것으로 장애물이 이동한다 하더라도 안전하게 로봇을 이동시킬 수 있으나[28] 비정형 미지의 환경에서는 이동 장애물들의 속도를 측정하는 것이

어려워서 사용 상의 어려움이 있다. 초기의 VFH 알고리즘은 초음파 센서로부터 획득되는 확실성 격자지도(certainty grid map)을 사용해서 만든 polar histogram 에 기초를 둔다.[29] VFH+ 알고리즘은 GuideCane에 사용하기 위해서 로봇의 크기와 속도를 고려해서 masked polar histogram[30]을 생성한다.

본 논문에서는 장애물 회피 알고리즘으로 GuideCane에 적용된 VFH+알고리즘을 변형하여 사용한다. 2D LRF로부터 측정된 데이터는 측정된 데이터는 VFH+ 알고리즘에서 지역 지도(local map) 생성을 위해서 쓰인다. VFH+ 알고리즘으로 방향각을 구하기 위해서는 지역 지도를 생성하고 이를 이용하여 Obstacle Vector를 만든 후에 Primary Polar Histogram, Binary Polar Histogram 그리고 Masked Polar Histogram을 차례로 생성한다. 최종적으로 생성된 Masked polar histogram과 목표 방향과 현재 로봇의 방향을 참고하여 방향각을 계산한다. 각각의 과정에 대해서 서술하면 다음과 같다.

#### 1) 지역지도 생성 (Creation of the Local Map)

지역지도는 로봇을 중심으로 로봇 주변의 장애물을 기록하는 격자지도다. 실험에서는 격자지도의 크기를 33x33 셀로, 하나의 셀은 10cm x 10cm 크기를 가지도록 설정했다.

#### 2) Obstacle Vector

격자지도의 활성 영역 내의 각 셀의 장애물이 존재할 확률 값으로부터 obstacle vector를 만든다.



### 3) Primary Polar histogram

로봇이 주행하는 영역에는 로봇 크기만큼의 공간이 필요하다. Local Map의 장애물 크기를 로봇의 크기로 늘이고 로봇은 하나의 점으로 생각한다. Primary polar histogram  $H^P$  에서 장애물이 영향을 미치는 범위를 계산하고, 장애물 확률을 업데이트 한다.

### 4) Binary Polar Histogram

이력현상(hysteresis)에 기반한 두 개의 한계값  $\tau_{low}$   $\tau_{high}$  를 사용하여 다음과 같은 조건으로 Binary polar histogram  $H^b$  를 생성한다.

$$H_{k,i}^b = \begin{cases} 1 & \text{if } H_{k,i}^P > \tau_{high} \\ 0 & \text{if } H_{k,i}^P < \tau_{low} \\ H_{k,i-1}^b & \text{otherwise} \end{cases}$$

### 5) Masked Polar Histogram

로봇이 빠른 속도로 전진하는 경우 좌우로 안전하게 회전하기 위한 최소 곡률반경이 존재하는데, 이 최소 곡률반경 과 장애물이 겹치게 되면 장애물이 있는 위치부터 로봇의 뒤쪽까지 로봇이 진행하지 못하도록 한다. 이를 반영하여 히스토그램을 수정한다.

### 6) Selection of the Steering Direction

Masked polar histogram에서 로봇이 지나갈 수 있는 모든 열린 부분을 찾는다. 열린 부분의 시작과 끝 섹터를  $k_l, k_r$  과 라고 할 때, 넓이가  $S_{max}$  보다 크면 다음 식으로 후보 섹터의

범위  $[c_l, c_r]$  을 정한다

$$c_l = k_r + \frac{S_{\max}}{2}, c_r = k_l - \frac{S_{\max}}{2}$$

넓이가 특정 값보다 작으면 다음 식으로 후보 방향  $c_l, c_r$  을 정한다

$$c_l = c_r = \frac{k_l + k_r}{2}$$

후보 섹터에 속하는  $c \in [c_l, c_r]$  에 대하여 평가함수  $g(c)$  를 아래와 같이 적용하여 최소 값이 되는 방향을 로봇의 방향으로 정한다.

$$g(c) = \mu_1 \Delta(c\alpha, \theta_t) + \mu_2 \Delta(c\alpha, \theta_i) + \mu_3 \Delta(c\alpha, \theta_{n,i-1})$$

위의 수식에서  $\theta_t$  는 목적지 방향,  $\theta_i$  는 로봇의 현재 방향 그리고  $\theta_{n,i-1}$  는 이전 상태에서 선택된 방향을 각각 나타낸다.

여기서  $\Delta(c_1, c_2)$  는 두 각도  $c_1, c_2$  사이의 최단거리 절대값을 계산하는 함수이다

$\mu_1, \mu_2, \mu_3$  는  $\mu_1 > \mu_2 + \mu_3$  조건이 만족하도록 설계하는데 본 논문의 실험 시에는 다음과 같이 설정했다.

$$\mu_1 = 5, \mu_2 = 1, \mu_3 = 0.5$$

최종적으로 생성된 masked polar histogram 과 타겟 방향 및 현재 방향을 참고하여 방향 각을 계산한다. 2D LRF 는 측정 영역이  $-135$  도 ~  $135$  도 이지만 실험에서는  $-90$  ~  $90$  도의 영역에서 측정되는 데이터만을 사용했다. 로봇의 크기는  $0.8m$  이며 지역 지도 크기는  $3 \times 3m$  로 정했다.

사용자와 0.8 미터를 유지하며 주어진 목표점으로 이동하도록 프로그램 했으며 IR 랜드마크 감지기로써 하기소닉 사의 스타게이저를 사용했다. 총 16 개의 IR 반사 물질로 구성된 마크를 천장에 붙였으며 높이는 2.5m 이고 수평으로 1m 씩 떨어지도록 배치했다.

방향각,  $h_{vfh}$  는 최종 Masked Polar Histogram에 기초해서 연산된다. 목표 방향 각은 간단히  $h_{vfh}$  와  $O_T$ 의 합으로 구해진다. 그리고 방향각에 대한 에러 값인  $h_e$  는 목표 각 과 현재 각 의 차이로 구해지며 아래의 수식과 같다.

$$h_e = (h_{vfh} + O_T) - h_c \quad (4.10)$$

위의 수식에서  $h_c$  는 IR 랜드마크 감지기로부터 구해진 현재의 방향각을 의미하며  $O_T$  는 Intelligent Lead로부터 생성된 목표 방향 변화량을 의미한다. 위치 에러  $p_e$  는 다음과 같이 계산된다.

$$p_e = (p_d + O_S) - O_D \quad (4.11)$$

위의 수식에서  $p_d$  는 사용자와 로봇간의 목표 간격으로써 0.8 미터로 정했다.  $O_S$  는 Intelligent Lead로 부터 생성된 목표 속도 변화량이며,  $O_D$  는 Intelligent Lead에 의해서 계산된 사용자와 로봇간의 거리가 된다.

최종 블록다이어그램은 다음과 같다.

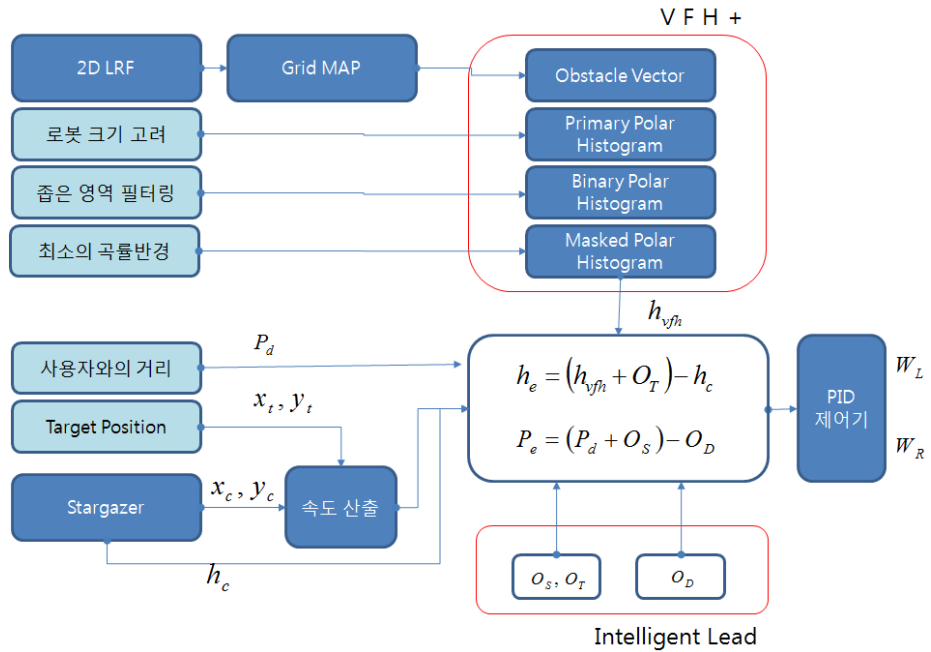


그림 4.5 로봇 모션 제어 알고리즘 블록다이어그램

위의 블록다이어그램에서 하늘색 블록들은 미리 정의된 상수들을 의미한다. 또한 파란색 블록은 센서에 의해서 연산되는 블록들을 나타낸다. 본 논문의 실험에서는 PID 컨트롤러는 PD 컨트롤러를 사용했다.

### 4.3 실험 시스템 구성

안내 로봇이 가져야 할 기본 기능으로 장애물 탐지 및 회피 기능을 들 수 있다. 실험 공간은 6x6미터 이며 중간 중간에 지름 50cm의 플라스틱 원통을 배치해서 장애물 회피 기능을 실험했다. 장애물

인식을 위해서 실험에 사용한 2D Laser scanner는 Hokuyo사의 UTM30-LX제품으로 아래의 표는 주요 특성을 나타낸다.


	Parameter	Typical value
	Light Source	Laser diode ( $\lambda = 785 \text{ nm}$ ), Class 1 (FDA)
	Detection Range	0.1 to 30 m, Max. 60 m
	Accuracy	0.1 to 10 m: $\pm 30 \text{ mm}$ , 10 to 30 m: $\pm 50 \text{ mm}$
	Scan 각	270°
	Angular Resolution	0.25° (360° /1,440 steps)
	Scan Time	25msec/scan
	Interface	USB2.0 (Full Speed)

그림 4.6 Hokuyo UTM-30LX 스펙

로봇의 현재 위치를 파악하는 방법으로 하기소닉사에서 개발한 스타게이저를 사용했다. 스타게이저는 천장에 적외선을 반사하는 재질로 된 ID를 지닌 랜드마크에 대한 로봇의 상대적인 위치와 방향을 적외선 이미지를 해석함으로써 고정밀도로 계산하는 원리를 이용한다. 스타게이저에 탑재된 적외선 프로젝트에서 방사된 적외선이 천정에 부착되어 있는 적외선 반사물질인 랜드마크에서 반사되면 스타게이저 내부에 장착된 렌즈를 통하여 영상으로 입력되며, 입력된 영상을 분석한다. 랜드마크는 위치(Position)와 방향(각)을 나타내는 정보와 ID 숫자를 가진다. 아래의 그림은 스타게이저의 하드웨어 부분과 랜드마크의 종류 및 형태를 나타낸다.

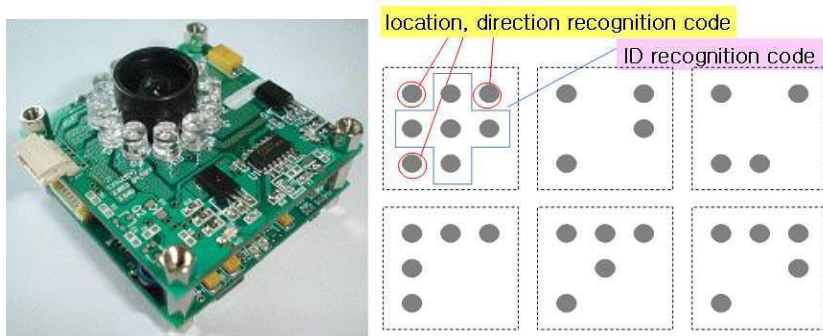


그림 4.7 스타게이저와 랜드마크

본 논문에서는 2D LRF에 의해서 측정되는 장애물 정보와 IR 랜드마크 감지기에 의해서 측정되는 로봇의 현재 위치에 의해서 VFH+ 알고리즘으로 로봇의 방향과 속도를 결정한다.

실험을 위해서 두개의 바퀴와 두개의 캐스터를 가진 형태의 이동 로봇을 제작했다. 미쯔비시 400W 용 AC 서보모터와 10 대 1 감속용 베벨 기어 박스 및 13 인치 바퀴를 사용했으며 내부에는 PC( intel i7-2600 CPU @3.4GHz)를 탑재했다. 서보모터 컨트롤러는 TI 사의 TMS320F2811 보드를 이용해서 제작했으며 PC로부터 RS232 통신에 의해서 왼쪽 바퀴와 오른쪽 바퀴에 대한 이동 제어량을 수신한다. 다음은 제작된 Intelligent Lead 와 테스트용으로 제작된 이동 로봇의 시스템 블록 다이어그램을 나타낸다.

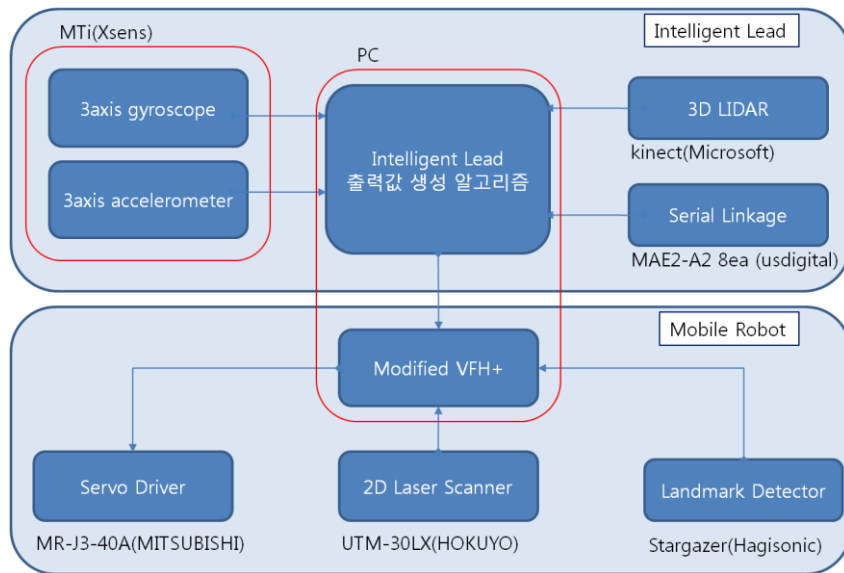


그림 4.8 안내 로봇 블록다이어그램

2D 레이저 스캐너로부터 획득되는 장애물 정보와 랜드마크 감지기로부터 획득되는 현재의 위치정보를 이용하여 VFH+ (Vector Field Histogram Plus) 알고리즘[8,9]에 의해서 로봇의 속도와 방향을 조절한다.

아래의 그림 4.9 는 로봇 앞면에서 찍은 사진으로 실험을 위해서 제작된 이동 로봇에 2D LRF 와 스타게이저가 부착되어 있는 것을 보여주며 그림 4.10 은 로봇 뒷면에서 찍은 사진으로 Intelligent Lead 가 부착되어 있는 것을 보여 준다.



그림 4.9 이동 로봇과 부착된 Intelligent Lead(앞면)

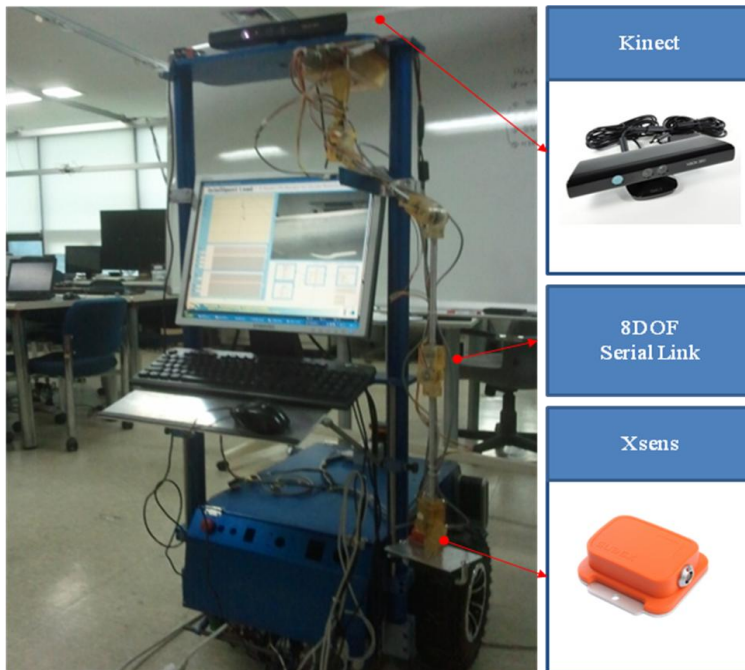


그림 4.10 이동 로봇과 부착된 Intelligent Lead (뒷면)



아래의 그림은 실험을 위해서 제작한 모니터링 프로그램 (monitoring program)을 나타낸다.

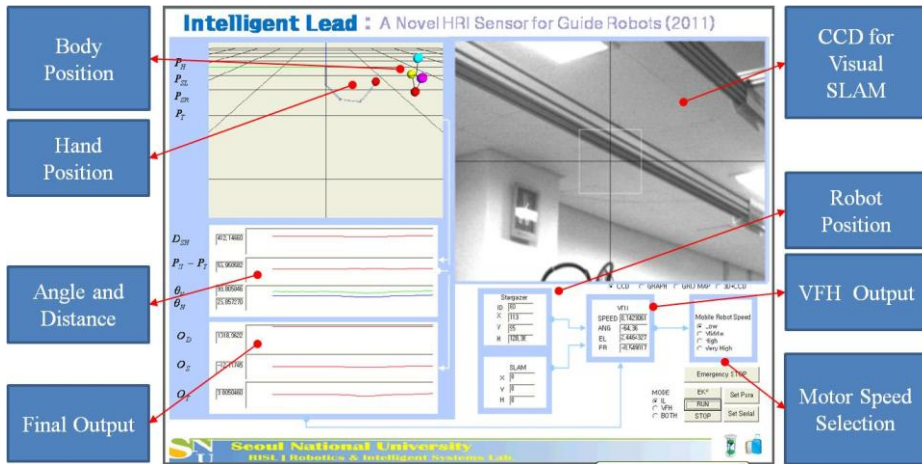


그림 4.11 모니터링 프로그램

좌 상단에 OpenGL 을 사용하여 시리얼 링크와 사용자로부터 획득된 머리, 왼쪽 어깨, 오른쪽 어깨, 몸체의 위치를 3 차원으로 나타내었으며 좌 하단에는 위의 4 개의 위치로부터 구해진 값들을 표시했다.

## 제 5 장 실험결과

이번 장에서는 제작된 Intelligent Lead를 실험을 위해서 제작된 안내 로봇에 부착하고, 4장에서 기술한 Intelligent Lead의 출력값 생성 알고리즘과 이동 로봇의 모션 제어 알고리즘을 다양한 실험을 통해서 검증한다. 1.2 절의 문제정의 및 해결방법에서 서술한 바와 같이 로봇과 사용자간의 안전거리 유지, 장애물 회피 및 목표점으로의 이동과 같은 공동 작업이 가능하도록 필요한 센서 시스템과 로봇의 모션제어알고리즘을 구현하는 것을 목표로 하며 이것을 위해서는 제작된 안내 로봇은 다음과 같은 기본 기능을 가지고 있어야 한다.

- 1) 사용자의 방향과 속도 제어 명령을 추종하는 기능.
- 2) 미리 정해놓은 사용자와 로봇과의 거리를 정해진 보행속도 이내에서 스스로 유지시키는 기능.
- 3) 스스로 장애물을 탐지하고 회피하는 기능.

4 장에서는 위의 세 가지 기능을 수행할 수 있도록 하기 위해서 VFH+알고리즘을 변형 적용한 모션 제어 알고리즘에 관해서 서술했다. 본 장에서는 제안한 모션 제어 알고리즘을 검증하기 위해서 다음과 같은 다섯 가지 실험을 수행한다. 첫 번째 실험으로 Intelligent Lead 출력값 생성 부분에 대해서 기능을 검증한다.

원하지 않는 제어 의도를 필터링(filtering)하는 기능에 대한 검증이 핵심이 된다. 두 번째로 이동로봇의 장애물 회피 알고리즘을 실험하며 분석한다. 세 번째 실험으로 Intelligent Lead 를 모바일 조이스틱으로 사용하고 로봇을 사용자가 원하는 목표점으로 이끌고 가는 경우에 대해서 실험한다. 이때 스타게이저에 의해서 측정된 현재의 위치 값이 로봇의 모션제어 알고리즘으로 유입되지 않아서 로봇의 위치 제어 알고리즘은 사용자와의 거리를 맞추는데 초점을 둔다. 네 번째로 스타게이저에 의해서 측정되는 위치 값이 제어 알고리즘으로 삽입되는 경우에 대해서도 테스트를 한다. 위의 경우에는 설정된 목표점과 현재의 위치가 PID 제어기에 반영된다. 따라서 모바일 조이스틱 실험에 비해서 목표점에 도달할 때의 에러가 훨씬 더 줄어들 것임을 예측할 수 있다. 위의 실험은 사용자와 로봇간의 협업 실험으로써 사용자가 리더가 되어서 로봇의 회전/속도 제어 명령으로 로봇을 원하는 위치로 이동시키며 로봇은 장애물 회피 및 사용자와의 거리 유지 기능을 담당하게 된다. 마지막으로 로봇이 사용자를 리드하는 실험으로 사용자가 속도/회전 제어 명령을 주지 않고 정해진 최대 속도 이내에서 로봇을 따라갈 때 로봇이 장애물 회피 및 목표점으로의 안내 기능을 담당할 수 있는 지에 관해 실험한다. 위의 실험은 안내건을 모사하는 실험으로 애완견과 산책을 할 때의 경우처럼 사용자는 Intelligent Lead 를 잡고 로봇 뒤를 따라가고 로봇은 장애물 회피 및 사용자와의 거리유지 기능을 수행한다.

## 5.1 센서 출력값 측정 및 분석

첫 번째 실험으로 로봇이 정지한 상태에서 Intelligent Lead를 수평으로 움직이며 출력되는 회전 제어 량과 속도 제어 량을 측정했다. [그림 5.1]은 속도 증가, 속도 감소, 왼쪽 회전, 오른쪽 회전을 명령을 각각 내렸을 때 획득된 사용자의 머리, 어깨, 몸체의 위치와 손목의 위치를 OpenGL을 이용하여 3D로 보여준 것이다.

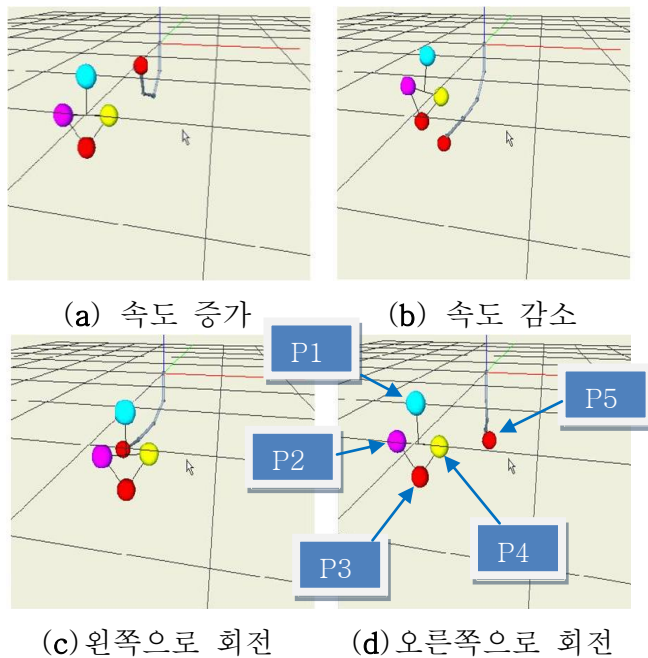


그림 5.1 핸들과 사용자의 관절 위치

그림 5.1은 손목을 흔들어서 위치에 변화를 주었을 때 측정된 값을 나타낸다. 위의 그림에서 P1은 사용자의 머리의 위치를, P2는

왼쪽 어깨의 위치, P3는 몸체의 위치, P4는 오른쪽 어깨의 위치 그리고 P5는 핸들의 위치를 나타낸다.

실험 중간에  $\theta_V$  가 미리 정한 한계 값인 45도를 경유하여 변화되도록 몸체를 기울였다. 손목의 높이가 몸체의 위치보다 낮았다가 높아지도록 한 후에 다시 높았다가 낮아지도록 테스트했다.

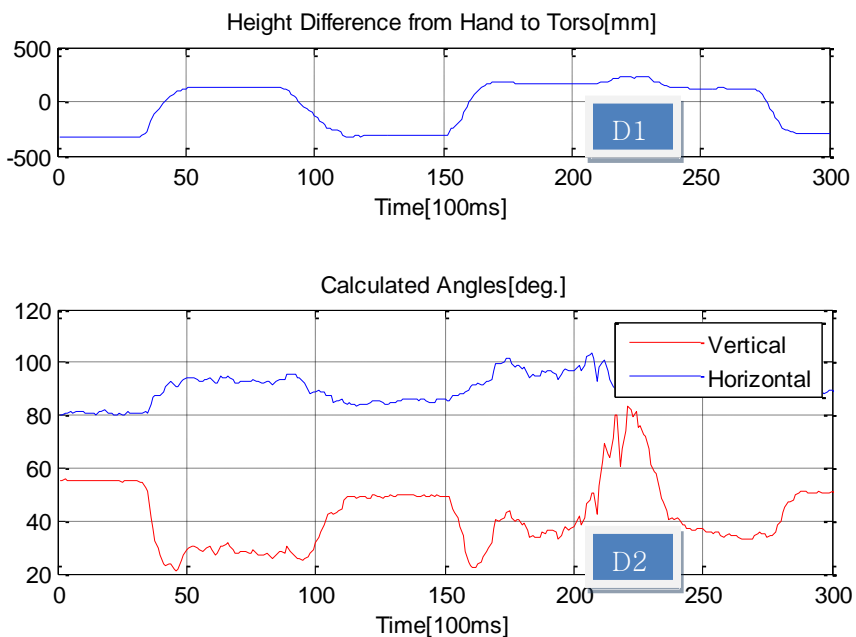


그림 5.2 Intelligent Lead 내부 연산값

그림 5.2에서 보여지듯이, time tick 200~250 영역에서 사용자가 허리를 굽힘으로써 손목의 높이가 몸체 높이보다 높아졌으며  $\theta_V$  가 45 도 보다 커지게 된다.

아래의 그림 5.3은 Intelligent Lead의 출력 값을 나타낸다.  $\theta_V$  가 45 도 보다 커지게 됨으로써 D2와 D3 구간에서 Intelligent

Lead 출력이 생성되지 않았지만 D1 구간에서는  $\theta_V$  값에 상관없이  
 사용자와 로봇간의 거리는 항상 연산되어 출력됨을 알 수 있다.

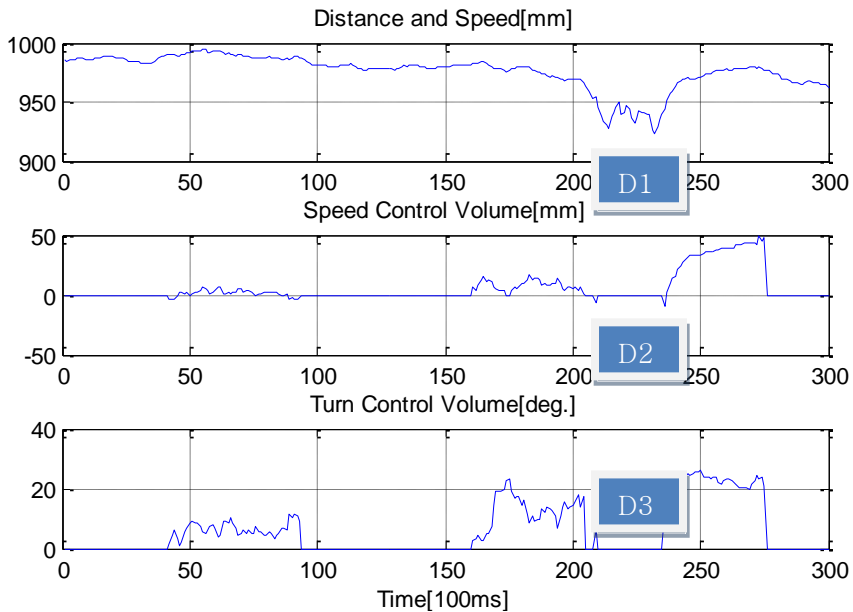


그림 5.3 Intelligent Lead 출력 값

위의 테스트로부터 사용자가 예기치 않은 상황에서 사용자가  
 물체를 기울였을 때 이것이 로봇의 제어 의도에 반영되지 않음을  
 확인할 수 있다.

## 5.2 장애물 회피 실험 및 분석

이번 실험은 로봇의 모션 제어 알고리즘에 삽입된 장애물 회피  
 알고리즘이 정상적으로 동작하는 지를 확인 하기 위한 실험이다.  
 또한 위의 실험을 통해서 목표점까지 도달할 때 에러가 어느 정도

발생하는 지에 대해서도 테스트한다. 이번 실험에서는 사용자는 전혀 관여하지 않고 Intelligent Lead의 출력 데이터도 로봇의 모션제어 알고리즘에 삽입되지 않는다.

로봇이 장애물을 회피하며 (1, 1.5), (1, 4.5), (4, 4.5) 그리고 (4, 1.5) 점들을 따라서 왕복 이동하도록 프로그램 했다. 장애물은 (2, 2.5), (2, 3.5), (3.5, 2.5) 그리고 (3.5, 3.5) 점들에 위치시켰다. 장애물들은 0.5미터의 지름을 가진 원통형 플라스틱이다.

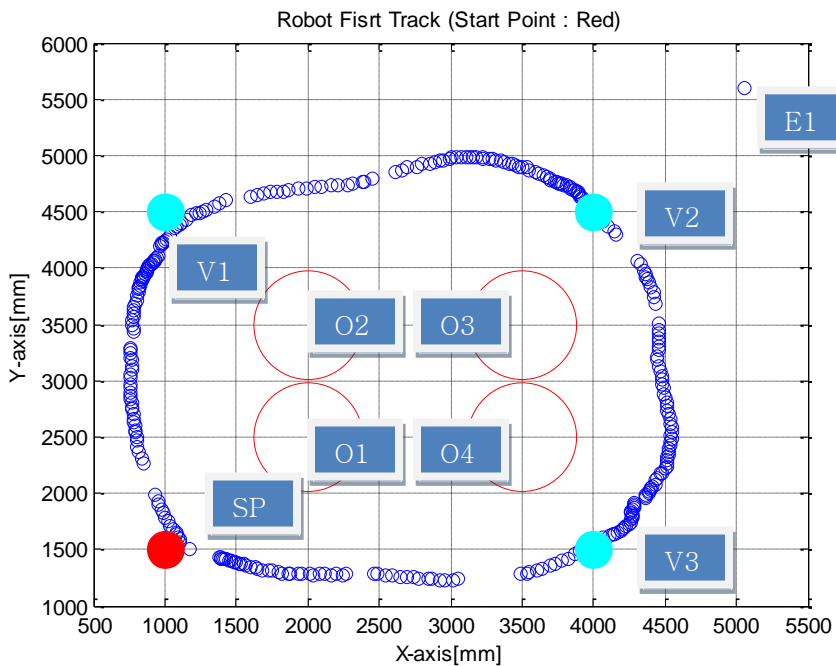


그림 5.4 로봇 트랙 - 장애물 회피 실험 1회

그림 5.4는 스타게이저로 측정한 로봇의 트랙을 보여준다. 위의 그림에서 SP는 “Start Point”를 V1, V2, V3는 “Via Point”를 O1, O2, O3, O4는 “Obstacle”을 의미한다. 그리고 E1은 스타게이저가 오류 데이터를 출력한 것으로 “Error”를 의미한다. 랜드마크들을

천정에 배치할 때 평탄도나 정렬에서 발생하는 에러로 인해서 스타게이지 내부의 카메라에 보여지는 랜드마크가 바뀔 때마다 로봇 패스가 끊기는 것을 볼 수 있다. 그럼에도 불구하고 로봇이 장애물들을 회피하며 원하는 목적지들에 도착함을 볼 수 있다.

아래의 그림은 한번 더 반복 수행했을 때 생성된 로봇의 트랙을 보여준다.

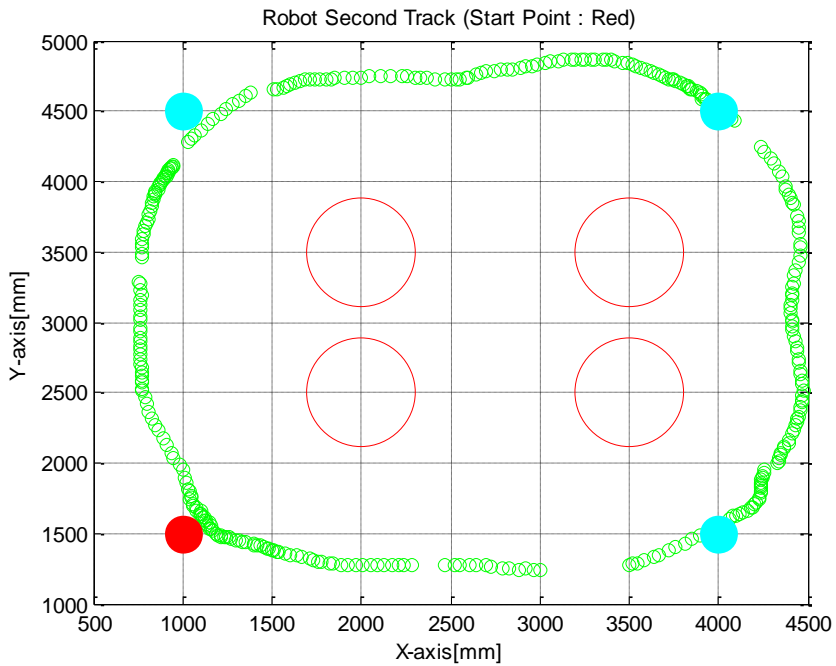


그림 5.5 로봇 트랙 - 장애물 회피 2회 실험

그림 5.5와 그림 5.6을 비교해보면 트랙의 휨 정도가 거의 같으며 휘어지는 부분들도 동일함을 알 수 있다. 스타게이지로 위치가 측정되지 않는 부분들도 동일함을 알 수 있다. 위의 실험에서 패스가 사각형의 형태를 형성하지 않는 이유는 장애물이



배치된 내부는 일정하지만 4개의 경유점을 기준으로 한 실험실의 바깥쪽 환경이 매우 복잡하기 때문이다. 바깥쪽에는 의자들이나 책상들이 그리고 각종 사무 집기들이 번잡하게 널려있다. 이러한 환경 정보가 레이저 스캐너로 측정되고 VFH알고리즘을 통해서 최적의 방향을 설정할 때 영향을 주었기 때문에 그림 5.6과 5.5와 같은 로봇의 패스가 형성되었다고 판단된다.

### 5.3 모바일 조이스틱 실험 및 분석

이번 실험은 로봇의 제어 알고리즘에 현재의 위치에 대한 정보를 삽입하지 않고 사용자의 회전과 속도 제어 명령에 의해서만 로봇이 움직이는 경우를 실험했다. 위의 실험은 상호간의 협업이 아니라 사용자가 Intelligent Lead 를 모바일 조이스틱(mobile joystick)으로써 사용할 수 있는 지에 대한 테스트가 된다. 본 논문에서는 조이스틱 패드가 이동하는 경우에 조이스틱 기능을 담당하는 센서를 “모바일 조이스틱” 이란 이름으로 사용했다. 로봇과 사용자가 동시에 움직일 때 고정된 조이스틱 패드 없이 조이스틱을 사용해서 로봇을 조종하는 것은 상당히 부자연스럽고 정밀하게 제어하기가 어렵다. 본 논문에서 제안하는 Intelligent Lead 는 외부의 다른 조이스틱을 사용하지 않고도 로봇을 원하는 위치로 옮기는 것을 가능하게 함으로써 사용자의 운용상의 편리함을

제공한다. 로봇의 모션제어 알고리즘 상에 Intelligent Lead 의 모든 출력 값들이 유입되며 로봇은 사용자와 0.8 미터를 유지하도록 프로그램 했다. 0.8 미터로 정한 것은 시리얼링크의 길이와 사용자의 보폭을 고려해서 시행착오를 통해서 잡은 것이다.

사용자는 (1, 1.5), (1, 4.5), (4, 4.5) and (4, 1.5) 점을 두 바퀴 도는 것을 목표로 한다. 처음 한 바퀴는 2 분(120 초)에 돌도록 하고 두 번째는 80 초에 돌도록 Intelligent Lead 의 속도 제어량을 조절하며 회전했다. 아래의 그림 5.6 은 실험을 통해서 생성된 로봇의 Track 을 보여준다 빨간 점은 시작점을 의미하며 초록 점들은 경유점들을 의미한다.

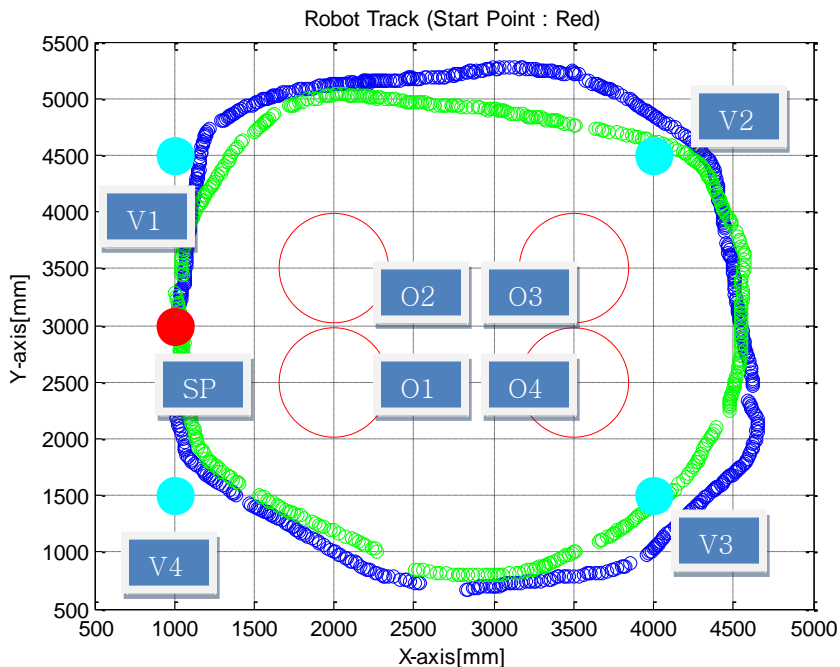


그림 5.6 로봇 트랙 - 모바일 조이스틱 실험

사용자와 로봇간의 거리의 평균은 0.8101 미터가 된다. 위의 실험으로부터 Intelligent Lead를 모바일 조이스틱으로 사용해서 사용자와의 거리를 유지하며 로봇을 원하는 목적지로 이동시킬 수 있음을 확인했다.

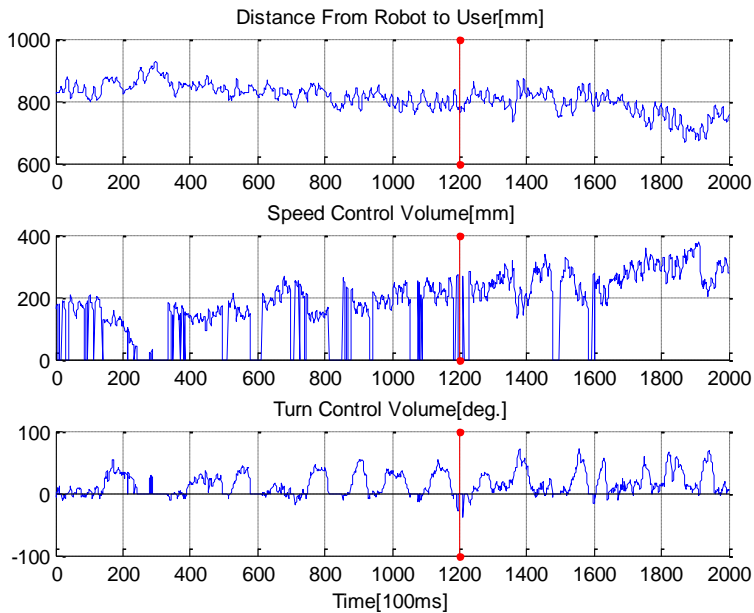


그림 5.7 Intelligent Lead 출력값 - 모바일 조이스틱 실험

시리얼 링크의 관절이 많아지고 전체 길이가 커지면 그만큼 더 여유로운 걸음새를 유지할 수 있게 된다.

위의 실험에서 목표점까지의 에러가 많이 생기는 이유는 바닥에 붙은 마크를 보고 조종하는 것이고 로봇 뒤에서 조종하기 때문이다. 정확한 위치로 인도하지는 못했으나 모바일 조이스틱으로 사용 가능함을 실험으로 검증했다.

## 5.4 사용자와 로봇의 협업 실험 및 분석

네 번째 실험으로 사용자와 로봇이 협업을 하며 장애물을 회피하며 목표점에 도달하는 테스트를 수행했다. 사용자는 Intelligent Lead를 이용해서 로봇에게 회전 제어 명령과 속도제어 명령을 줄 수 있으며 로봇은 Intelligent Lead로부터 유입되는 사용자와의 거리 정보를 모니터링하며 정해진 간격을 유지하는 기능을 수행한다. 또한 스타게이지로부터 유입되는 현재의 위치 및 방향과 미리 정해진 목표점과 차이를 이용해서 정해진 목표점으로 사용자를 이끌게 된다. 또한 2D 레이저스캐너로부터 유입되는 환경정보를 이용해서 장애물 회피를 위한 최적의 방향각을 계산한다. 따라서 사용자와 로봇 모두 회전 제어 명령과 속도 제어 명령의 주체가 되므로 상호간에 협업 관계를 가지게 된다. 본 논문에서는 간단히 PID 제어를 통해서 위의 기능을 구현했으며 4.2 절의 제어기의 블록다이어그램에 명시되어 있다. 로봇이 사용자와 0.8미터의 거리를 유지하며 사용자를 이끌도록 프로그램 했다. 장애물의 영향을 테스트하기 위해서 장애물들을 아래에 배치하고 (1, 1.5), (1, 4.5), (4, 4.5) 그리고 (4, 1.5) 포인트를 경유하여 이동하는 실험을 수행했다. 아래의 그림 5.8은 이때 생성된 로봇 트랙을 나타내며 그림 5.9는 Intelligent Lead의 출력값을 나타낸다.

앞의 실험보다 더 정확하게 정해진 목표점에 도달하는 것을 알 수 있다.

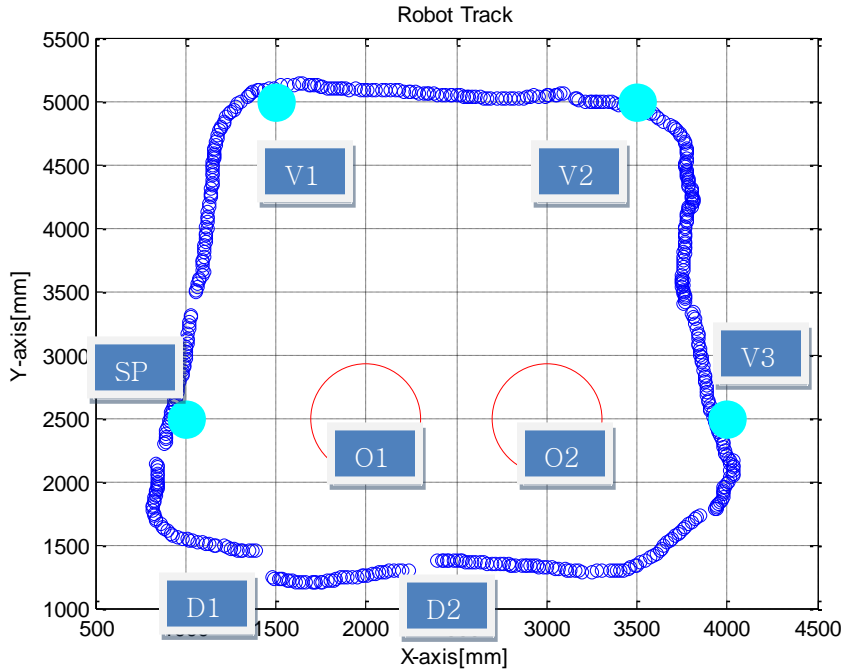


그림 5.8 로봇 트랙 - 사용자와 로봇의 협업 실험

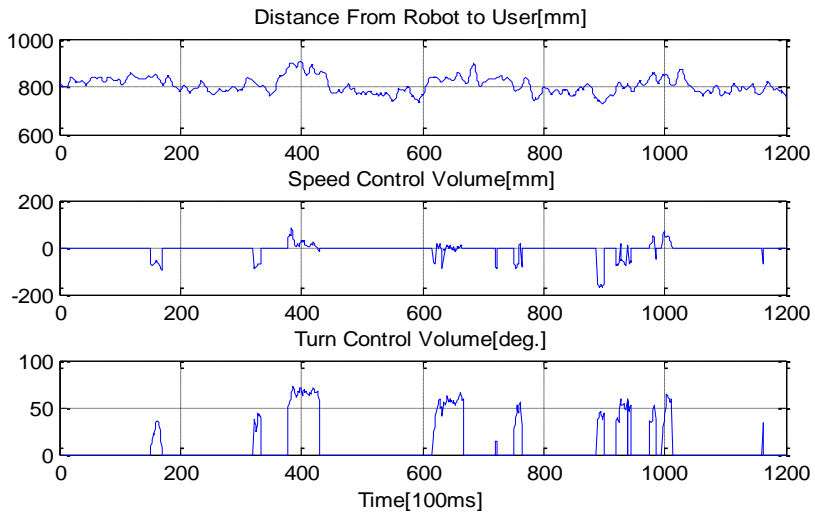


그림 5.9 Intelligent Lead 출력값 - 협업 실험

로봇과 사용자의 평균 거리는 0.806677 미터이고 최대 값은 0.907718 미터이며 최소값은 0.727042 미터가 된다. 그림 5.8에서 두 부분에게 스타게이저의 데이터가 끊긴 것은 천정에 붙은 랜드마크가 정확히 수평으로 붙여지지 않았기 때문으로 판단된다. 또한 그림 5.9를 보면 로봇이 회전할 때 사용자와의 간격이 증가하거나 감소하는 경향을 보이며 일정한 방향으로 진행할 때는 간격에러가 현저히 줄어드는 것을 볼 수 있다. 또한 사용자의 제어 명령이 앞의 실험 결과인 그림 5.7보다 훨씬 단조로운 것을 볼 수 있다. 이것은 그만큼 사용자가 덜 피곤했음을 의미하며 목표점까지도 정확하게 도달하는 것을 확인할 수 있다. 이것은 제안한 제어기의 성능이 우수함을 의미한다.

## 5.5 안내견 모사 실험 및 분석

마지막 테스트로 안내견을 모사하는 실험으로써 사용자의 도움 없이 안내 로봇이 사용자를 리드(lead)하며 원하는 목적지로 도달하는 테스트를 수행했다. 애완견과 산책을 할 때의 경우처럼 사용자는 Intelligent Lead 를 잡고 로봇 뒤를 따라가고 로봇은 장애물 회피 및 사용자와의 안전 거리유지 기능을 수행한다.

이 실험의 목적은 안내 로봇이 스스로 장애물을 인지하고 목표점까지 사용자의 보행 속도에 맞추어서 사용자와의 거리를

정해진 간격만큼 유지하면서 사용자를 목표점으로 안내할 수 있는 안내 로봇 개발을 목표로 로봇 모션제어 알고리즘의 성능테스트를 위한 것이다. 아래의 그림은 장애물의 위치와 경유점들 및 진행방향을 나타낸 것이다.

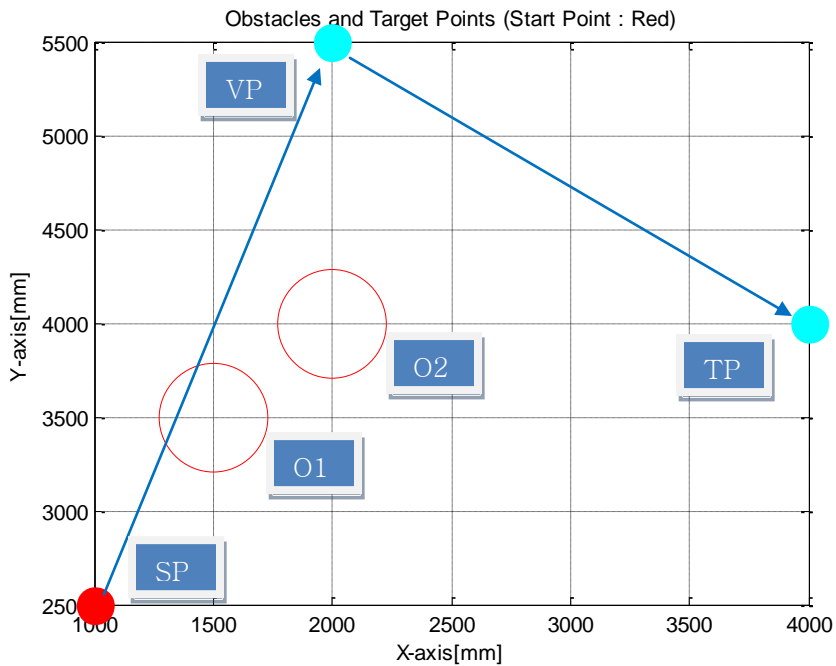


그림 5.10 경유점 및 장애물의 위치

위의 그림에서 SP 는 Start Point 를 의미하며, O1, O2 는 Obstacle 들을 의미한다. VP 와 TP 는 각각 Via Point 와 Target Point 를 의미한다. 그림 상에 보이는 것처럼 로봇은 (1, 2.5), (2, 5.5) and (4, 4) 점을 따라가고 이전의 실험에서처럼 사용자와의 거리를 0.8 미터로 유지하도록 프로그램 했다. VP 와 SP 를 잇는 선분이 O1 과 겹치도록 즉, 목표점으로 향하는 직선상에 장애물을

배치함으로써 장애물을 만났을 때 어떤 방식으로 로봇이 사용자를 안내하는 지에 대해서 경향을 분석하고자 했다.

사용자는 Intelligent Lead 의 핸들을 잡고 로봇과의 거리에는 신경 쓰지 않고 로봇이 이끄는 방향을 따라간다. 사용자가 걸을 수 있는 최대 속도는 1m/s 로 설정했다. 실시간으로 구해지는 사용자와의 거리로부터 사용자의 걸음속도를 연산하며 최대 속도 이상으로 이동하면 로봇은 정지하도록 프로그램 했다. 최대 속도 이내에서는 사용자의 걸음속도에 맞추어 사용자와의 거리를 정해진 목표 간격으로 로봇이 스스로 유지하는 실험이 된다.

제어 알고리즘 상에서 회전 제어 량은 장애물 정보와 목적지 정보 그리고 스타게이저로부터 측정되는 현재의 위치 및 로봇의 방향 정보로부터 획득된다.

이번 실험에서 사용자는 Intelligent Lead 를 통해서 회전 제어 명령과 속도제어 명령을 로봇에게 전달하지 않고 편한 자세로 Lead 를 붙잡고 걸으며 로봇이 방향 및 속도를 제어하며 사용자와의 거리를 유지하는 기능을 담당한다. 또한 로봇은 장애물에 대한 정보를 실시간 파악하고 장애물 회피 알고리즘을 사용하여 장애물을 피하면서 목표점으로 이동하는 목표를 가지고 있다. 아래의 그림은 실험을 통해서 획득된 로봇의 이동 궤적을 나타낸다.



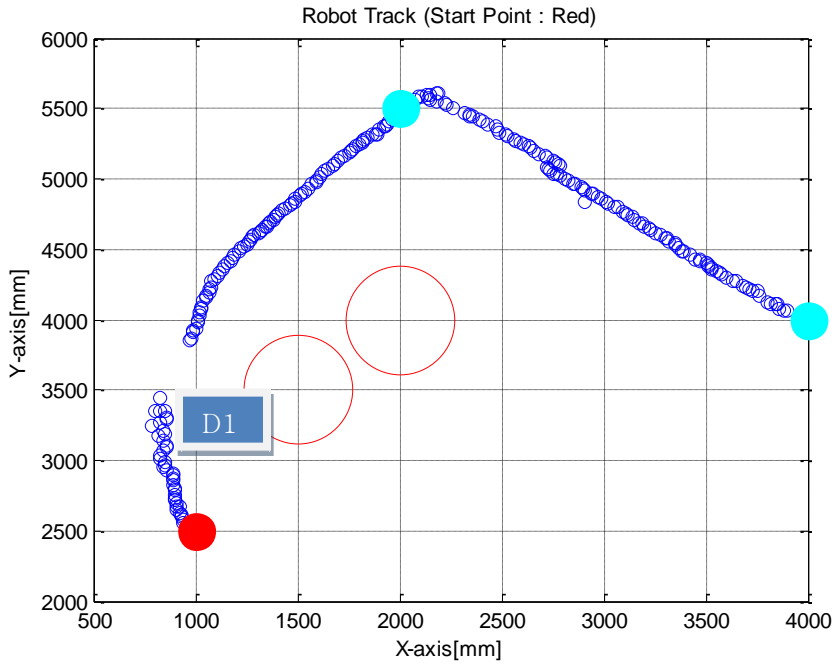


그림 5.11 로봇 패스 - 안내건 모사 실험

위의 그림에서 초기 위치는 (1.5, 1.3)이며 첫 번째 경유점인 (2.5, 1.0) 점으로 이동했다. (1.5, 3.5)점에 위치한 장애물에 의해서 좌측으로 미세하게 휘어지며 호를 그리는 것을 볼 수 있다. 다음 목표점인 (2.0, 5.5) 점을 향해 진행하는데 직선 상에 장애물이 위치하므로 호를 그리며 장애물을 회피하는 것을 볼 수 있다. (0.8, 3.5) 점 근처에서 즉, 그림상에서 D1 부분에서 패스가 끊어지며 점들이 산만하게 배치되는 것이 확인되는데 이것은 로봇이 급박하게 움직임으로써 로봇의 자세가 흔들리고 로봇의 흔들림이 스타게이저 센서에 영향을 주었기 때문으로 해석된다. 스타게이저는 측정 카메라가 심하게 흔들리거나 랜드마크들과

수평을 유지하지 못하면 위치 에러가 증가되며 심한 경우에는 랜드마크를 찾지 못하거나 오류 데이터를 출력하는 경우가 자주 발생한다. 두번째 경유점에 도착한 후에 마지막 목표점까지는 장애물이 없으므로 직선을 따라서 로봇이 이동한 것을 볼 수 있다. 로봇이 VFH+알고리즘을 사용하여 장애물을 회피함과 동시에 미리 정해놓은 목표점을 향해가고 사용자와의 일정거리를 유지하는 이번 테스트의 결과그래프를 볼 때, 본 논문에서 새롭게 제안된 Intelligent Lead 를 안내 로봇의 위치측정센서로써 사용하는 것이 적절함이 검증되었다.

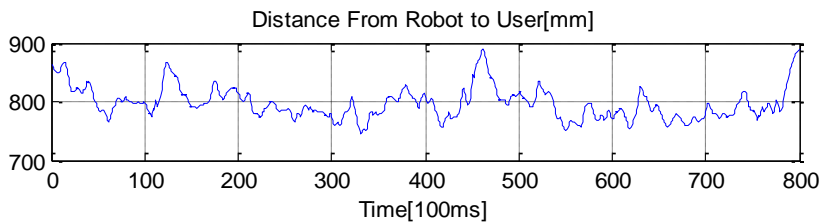


그림 5.12 Intelligent Lead 출력값 - 안내견 모사 실험

그림 5.11 에서 로봇이 이전에 정한 목적 점들을 장애물을 회피하며 따라가고 있음을 확인했다. 그림 5.12 는 실험하는 동안에 측정된 Intelligent Lead 의 출력값 들 중에서 사용자와 로봇간의 거리를 그린 그래프이며, 사용자와의 평균 유지 거리는 0.796902 미터이고 최대는 0.890403 미터 그리고 최소는 0.74513 미터로 측정되었다.

이번 실험에서는 Intelligent Lead 의 회전 제어 량과 속도 제어 량을 사용하지 않고 사용자와 로봇간의 위치 데이터 만을 사용했으나 사용자의 손목의 움직임으로부터 사용자의 이동 방향에 대한 예측을 하는 블록을 마련하고 모션제어 알고리즘 상에 추가한다면 로봇이 더욱 부드러운 움직임을 보일 것으로 예측된다. 여러 번의 테스트 동안에 사용자의 몸체가 카메라의 시야각 영역을 벗어나서 OpenNI 알고리즘이 사용자의 관절정보를 획득하지 못하는 경우가 발생했다. 향후에는 위의 문제를 해결하기 위해서 얼굴인식 알고리즘을 삽입하고 사용자의 팬(pan)/틸트(tilt) 제어기를 통해서 사용자의 얼굴을 따라가는 구조를 취하면 위의 문제가 해결될 것으로 판단된다.

## 5.6 실험 결과 비교 정리

4 가지 실험에 대한 결과를 요약 비교하면 다음과 같다.

[장애물 회피 실험의 경우]

- 1) Intelligent Lead 를 사용하지 않고 로봇 모션 제어 알고리즘 성능을 평가했다.
- 2) 매끄러운 곡선으로 나타나지 않는 것은 의자나 사무집기류에 의해서 바깥 환경이 울퉁불퉁한 것이 VFH 알고리즘에 반영되었기 때문이다.

- 3) 두 번의 실험에서 트랙의 휨 정도가 거의 같으며 휘어지는 부분들도 동일함이 확인된다.

[모바일 조이스틱 실험의 경우]

- 1) Intelligent Lead 를 모바일 조이스틱으로 사용 가능한 지를 테스트했다.
- 2) 로봇에게는 사용자와 0.8 미터 거리 유지 기능이 부여되었다.
- 3) 목표점까지의 에러가 많이 생기는 이유는 바닥에 붙은 마크를 보고 조종하며 사용자가 로봇 뒤에서 조종하기 때문에 마크가 잘 안보이는 경우가 많았기 때문이다.
- 4) 정확한 위치로 인도하지는 못했으나 Intelligent Lead 를 모바일 조이스틱으로 사용가능함을 확인했다. 위의 실험 과정에서 평균 유지 거리는 0.8101 미터였다.

[사용자와 로봇의 협업 실험의 경우]

- 1) 사용자와 로봇이 공동으로 목표점 이동 및 장애물 회피의 주체가 되는 실험이다.
- 2) 로봇에게는 사용자와 0.8 미터 거리 유지 기능이 부여되었다.
- 3) 사용자의 시각적 정보가 Intelligent Lead 를 통해서 회전/속도 제어 명령으로 출력되었다.
- 4) 평균 유지 거리 0.8067 미터, 최대 거리 0.9977 미터, 최소 거리 0.7270 미터가 되었다.

[안내견 모사 실험의 경우]

- 1) 사용자가 핸들을 잡고 따라가는 경우를 실험했다.

- 2) 로봇에게는 사용자와 0.8 미터 거리 유지 기능이 부여되었다.
- 3) 사용자의 회전/속도 제어 명령이 모션제어기에 유입되지 않았다.
- 4) 사용자의 피로도를 최대한 줄이는 것을 목적으로 했다.
- 5) 평균 유지 거리 0.7969 미터, 최대 거리 0.8904 미터, 최소 거리 0.7451 미터로 측정되었다.

위의 실험들로부터 획득된 트랙데이터와 입력 변수를 하나의 표로 나타내면 다음과 같다.

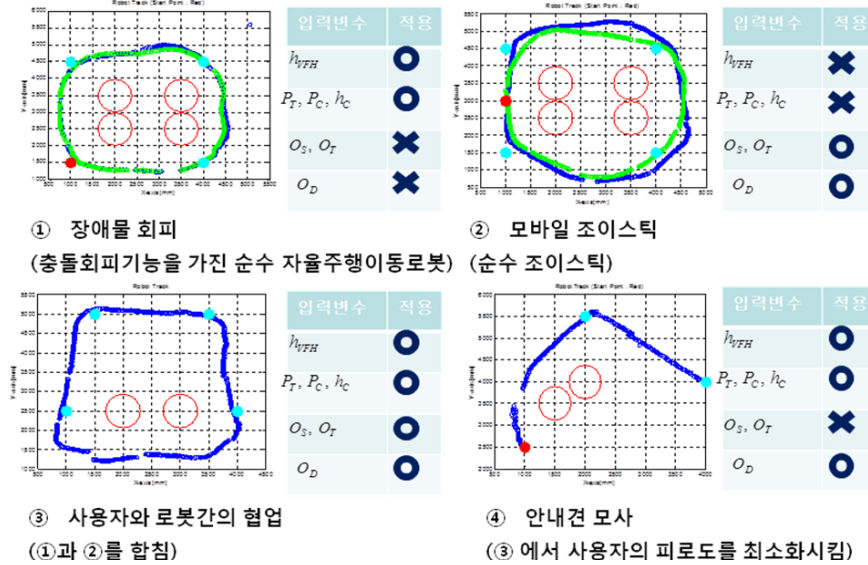


그림 5.13 입력 변수 적용 표와 로봇 패스 비교

안내견 로봇 제작을 위해서는 안내견 모사 실험이 필수적이며, 본 논문에서 제안하는 Intelligent Lead 와 로봇 제어 알고리즘으로 사용자의 피로도를 최소화하면서 원하는 장애물을 회피하며 원하는 목적지에 도달하는 안내견 로봇 제작이 가능함을 위의 그림 5.13 에서 확인 할 수 있다.

## 제 6 장 결론 및 향후 연구 방향

### 6.1 결론

본 논문에서는 외란의 영향을 중요시해야 하는 로봇들이 사용하기에 적합한 새로운 위치 측정 시스템으로, 관절각 또는 링크각을 측정할 수 있는 시리얼 링크의 한 끝에 IMU 를 부착해서 사용자와 로봇간의 거리와 사용자의 제어 의도가 반영된 회전 및 속도 제어량을 실시간 출력하는 Intelligent Lead 를 제안하고, 엔코더와 IMU 그리고 시리얼 링크로 구성된 샘플을 제작했으며, Intelligent Lead 의 성능 검증을 위해서 장애물을 회피하며 지정된 목표점으로 이동 가능한 안내 로봇을 제작했다.

또한 로봇의 모션 제어 알고리즘을 제안하고 장애물 회피 실험과 모바일 조이스틱 실험, 협업 실험 및 안내건 모사 실험을 통해서 Intelligent Lead 가 사용자의 회전 제어 명령, 속도 제어 명령의 출력 수단으로써 사용될 수 있을 뿐만 아니라 사용자와 로봇 사이의 거리를 안정적으로 출력함으로써 사용자의 자유로운 보행에 방해가 되지 않으면서도 사용자를 목적지까지 이끌어주는 안내건 로봇의 제작 및 상품화 가능성을 확인했다.

## 6.2 논문의 기여도

본 논문의 기여도는 다음과 같이 요약된다.

1) 시리얼 링크를 이용한 위치측정시스템으로써 Intelligent Lead를 안내 로봇 제작에 적용하는 것을 제안했다.

2) Intelligent Lead와 연동되어 사용할 수 있는 안내 로봇의 제어 알고리즘을 제안했다.

3) 안내 로봇과 Intelligent Lead를 제작하고 장애물 회피 실험과 모바일 조이스틱 실험과 사용자와 로봇간의 협업 실험과 안내건 모사실험을 통해서 Intelligent Lead의 성능을 검증했으며 안내건 로봇의 상용화 가능성을 실험으로 확인했다.

4) 관절각 측정 방식이 아닌 링크각 측정 방식에 의한 위치 측정으로 보다 더 소형화, 경량화 된 센서 시스템을 제작함으로써 사용 편의성을 향상시키는 방법을 제안했다.

## 6.3 향후 연구 방향

LIDAR의 시야각이 사람에 비해서 약 1/3정도밖에 되지 않으므로 안내 로봇이 장애물을 회피하기 위한 목적이나 순간적인 목표점의 변화로 인해서 급격한 회전을 하면 사용자가 시야각에서 벗어남으로써 사용자의 어깨나 몸체의 위치를 측정하지 못하는 문제점이 있다. LIDAR를 회전시키는 서보 시스템과 얼굴인식

소프트웨어 모듈을 삽입해서 이러한 문제를 해결 하는 것이 필요하다.

또한 안내 로봇이 랜드 마크의 의존성을 없애고 비정형 환경에서 사용될 수 있도록 하기 위해서 LRF 기반의 2D SLAM 이나 LIDAR나 비전 기반의 3D SLAM 알고리즘을 적용하고 현재 구현된 로봇 제어 알고리즘과 연동하는 실험이 필요하다.

마지막으로 시리얼 링크와 IMU 센서의 조합은 비행 로봇의 호버링(hovering)을 위한 새로운 위치 측정 시스템으로써 사용 가능하다. 실외에서 비행하는 비행 로봇에 탑재된 GPS는 건물 외벽이나 나무들과 같은 복잡한 주변환경에 의해서 성능이 저하된다. 또한 비전 기반의 SLAM 알고리즘은 단조로운 주변환경에서 특이점들의 추출량이 적어지거나 급격히 변하는 환경에서 특이점 정합량이 적어지는 문제점을 가지고 있다. 이런 센서들의 성능 저하로 인해서 자동제어가 어려워지거나 연무나 안개에 의해서 원격 조종이 불가능 할 때 사용자가 비상 버튼을 누르면 비행체를 착륙시키지 않고 호버링이 가능하도록 하기 위한 새로운 위치 측정 센서로써 Intelligent Lead를 사용하는 연구를 수행할 계획이다.



## 참고 문헌

- [1] V. Kulyukin, C. Gharpure, and J. Nicholson, "RoboCart: Toward Robot-Assisted Navigation of Grocery Stores by the Visually Impaired," in *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 2845-2850, Edmonton, Canada, Aug. 2005.
- [2] V. Kulyukin, C. Gharpure, J. Nicholson, and S. Pavithran, "RFID in Robot-Assisted Indoor Navigation for the Visually Impaired," in *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 1979-1984, Sendai, Japan, Sep. 2004.
- [3] V. Pradeep, G. Medioni, and J. Weiland, "Robot Vision for the Visually Impaired," in *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, pp. 15-22, San Francisco, California, USA, Jun. 2010.
- [4] I. Ulrich and J. Borenstein, "The GuideCane—Applying Mobile Robot Technologies to Assist the Visually Impaired," *IEEE Transactions on Systems, Man, and Cybernetics—Part A: Systems and Humans*, vol. 31, no. 2, pp. 131-136, Mar. 2001.
- [5] I. Shim, J. Yoon, and M. Yoh, "A Human Robot Interactive System—RoJi," *International Journal of Control, Automation, and Systems*, vol. 2, no. 3, pp. 398-405, Sep. 2004.
- [6] A. A. Melvin, B. Prabu, R. Nagarajan, and B. Illias, "ROVI: A Robot for Visually Impaired for Collision-Free Navigation," in *Proceedings of International Conference on Man-Machine Systems*, pp. 3B51-3B56, Penang, Malaysia, Oct. 2009.

- [7] G. Galatas, C. McMurrough, G. L. Mariottini, and F. Makedon, “eyeDog: An Assistive-Guide Robot for the Visually Impaired,” in *Proceedings of International Conference on Pervasive Technologies Related to Assistive Environments*, Crete, Greece, May 2011.
- [8] I. Ulrich and J. Borenstein, “VFH+: Reliable Obstacle Avoidance for Fast Mobile Robots,” in *Proceedings of IEEE International Conference on Robotics and Automation*, pp. 1572-1577, Leuven, Belgium, May 1998.
- [9] O. Ringdahl, “Path Tracking and Obstacle Avoidance Algorithms for Autonomous Forest Machines,” *Master Thesis*, Umea University, Sweden, 2003.
- [10] V. Kulyukin, C. Gharpure, and N. D. Graw, “Human-Robot Interaction in a Robotic Guide for the Visually Impaired,” in *Proceedings of AAAI Spring Symposium on Interaction between Humans and Autonomous Systems over Extended Operation*, Palo Alto, CA, USA, Mar. 2004.
- [11] J. A. Hesch, F. M. Mirzaei, G. L. Mariottini, and S. I. Roumeliotis, “A 3D Pose Estimator for the Visually Impaired,” in *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 2716-2723, St. Louis, USA, Oct. 2009.
- [12] H. K. Khalil, *Nonlinear Systems*, Prentice Hall, pp.111-133, 2001.
- [13] J. C. Latombe, *Robot Motion Planning*, Kluwer Academic Publishers, pp.295-313, 1991.
- [14] H. Choset, K. M. Lynch, S. Hutchinson, G. Kantor, W. Burgard, L. E. Kavraki, and S. Thrun, *Principles of Robot Motion*, The MIT Press, pp.77-90, 2005.

- [15] J. H. Lee, J. S. Choi, B. H. Lee, and K. W. Lee, "Complete coverage path planning for cleaning task using multiple robots," in *Proceedings of IEEE International Conference on Systems, Man and Cybernetics*, pp. 3618-3622, San Antonio, TX , USA, Oct. 2009.
- [16] J. Xiong and G. Tan, "Virtual Forces Based Approach for Target Capture with Swarm Robots," in *Proceedings of Chinese Control and Decision Conference*, pp. 642-646, Guilin, China, Jun. 2009.
- [17] Y. Koren and J. Borenstein, "Potential Field Methods and Their Inherent Limitations for Mobile Robot Navigation," in *Proceedings of IEEE International Conference on Robotics and Automations*, pp.1398-1404, Sacramento, CA, USA, Apr. 1991.
- [18] S. S. Ge and Y. J. Cui, "New Potential Functions for Mobile Robot Path Planning," *IEEE Transactions on Robotics and Automation*, vol.16, no.5, pp.615-620, Oct. 2000.
- [19] Website : [www.who.int](http://www.who.int)
- [20] John H. Blakelock, *Automatic control of Aircraft and Missiles*, WILEY-INTERSCIENCE
- [21] M. S. Grewal, L. R. Weill, A. P. Andrews, *Global Positioning Systems, Inertial Navigation, and Integration*, WILEY-INTERSCIENCE
- [22] K. S. Fu, R. C. Gonzalez, C. S. G. Lee, *Robotics Control, Sensing, Vision, and Intelligence*, McGRAW-HILL INTERNATIONAL EDITIONS
- [23] John J. Craig, *Introduction to Robotics mechanics and control*, Addison Wesley

- [24] Yoshihiko Nakamura, *Advanced Robotics Redundancy and Optimization*, Addison Wesley
- [25] O. Khatib, "Real-time Obstacle avoidance for manipulators and Mobile Robots," *IEEE International Conference on Robotics & Automation*, pp. 500–505, St. Louis, Missouri, 1985.
- [26] O. Khatib, "Real-time Obstacle avoidance for manipulators and Mobile Robots," *IEEE International Conference on Robotics and Automation*, pp. 500–505, St. Louis, Missouri, 1985.
- [27] D.An, H. Wang, "VPH: a new laser radar based obstacle avoidance method for intelligent Mobile Robots," *IEEE Int. Conf. on Intelligent Control and Automation*, Vol. 5, pp. 4681-4685, June 2004.
- [28] P. Fiorini and Z. Shiller, "Motion planning in dynamic environments using velocity obstacles," *International Journal of Robot. Res.* 17 (7), pp. 760–772, 1999.
- [29] Borenstein, J. and Koren, Y, "The Vector Field Histogram - Fast Obstacle Avoidance for Mobile Robots," *IEEE Journal of Robotics and Automation*, Vol. 7, No. 3, pp. 278-288, June 1991.
- [30] Ulrich, I. and Borenstein, J, "VFH+: Reliable Obstacle Avoidance for Fast Mobile Robots," *IEEE International Conference on Robotics and Automation*, pp 1572-1577, May 1998.
- [31] Ulrich, I. and Borenstein, J, "VFH\*: Local Obstacle Avoidance with Look-Ahead Verification," *IEEE International Conference on Robotics and Automation*, pp.2505-2511, 2000.

- [32] J. Gong, Y. Duan, Y. Man, G. Xiong, "VPH+: An Enhanced Vector Polar Histogram Method for Mobile Robot Obstacle Avoidance," *Proceedings of the 2007 IEEE International Conference on Mechatronics and Automation*, 2007.
- [33] Shi Dongqing, Dunlap Damion, Collins Emmanuel G., "A comparison between a fuzzy behavioral algorithm and a vector polar histogram algorithm for mobile robot navigating," *Proceedings of International Symposium on Computational Intelligence in Robotics and Automation*, pp. 260-265, 2007.

## Abstract

# A New Positioning System for Guide Robot Control

Cho, Keum Bae

School of Electrical Engineering and Computer Sciences

The Graduate School

Seoul National University

This research addresses the introduction of a new Human Robot Interaction (HRI) sensor for guide robots. Guide robots for geriatric patients or the visually impaired should follow user's control command, keeping a certain desired distance allowing the user to work freely. Therefore, it is necessary to acquire control commands and a user's position on a real-time basis. We suggest a new sensor fusion system to achieve this objective and we will call this sensor system the "Intelligent Lead". The objective of the intelligent lead is to acquire a stable distance from the user to the robot, speed-control volume and turn-control volume, even when the robot platform with the intelligent lead is shaken on uneven ground. In this research we have used a precise Extended Kalman Filter (EKF) procedure for this objective. The intelligent lead physically consists of a Kinect sensor, the serial link attached with eight rotary encoders, and an IMU (Inertial Measurement Unit), and their measurements are fused by the EKF. A mobile robot was

designed to test the performance of the proposed sensor system. After installing the intelligent lead in the mobile robot, several tests are conducted to verify that the mobile robot with the intelligent lead is capable of achieving its goal points while avoiding obstacles and maintaining the appropriate distance between the robot and the user. The results show that we can use the intelligent lead proposed in this paper as a new HRI sensor joined a joystick and a distance measure in the mobile environments such as the robot and the user are moving at the same time.

**Keywords :** Positioning system, Guide robot, Serial link, LIDAR, EKF

**Student Number :** 98420–868