

2022
ISSUE PAPER

자율 주행 자동차의 요소 기술 동향

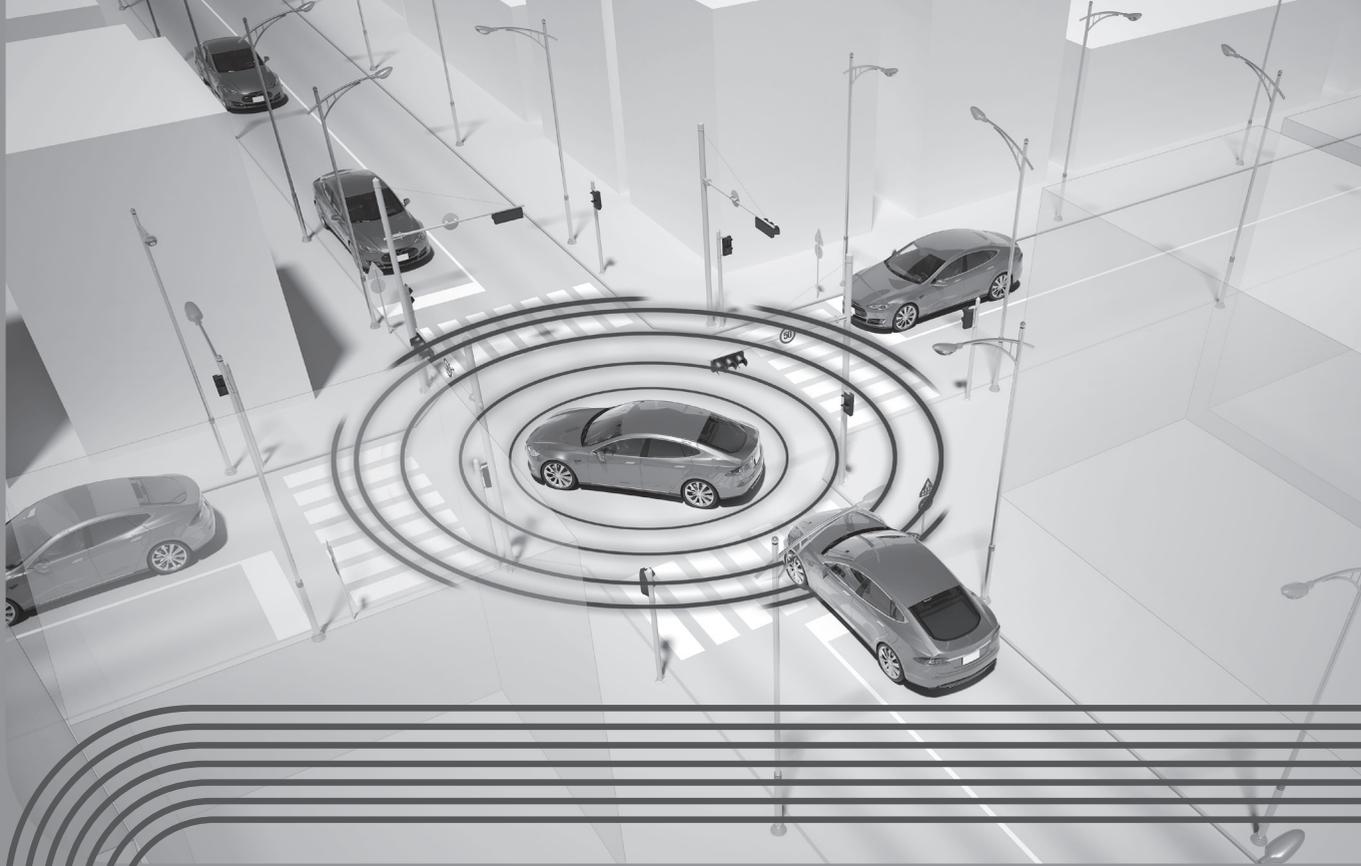
영진전문대학교 김재완 교수



경상남도
GYEONGNAM



재단법인 경남테크노파크
과학기술에너지센터



2022
ISSUE PAPER

자율 주행 자동차의 요소 기술 동향

영진전문대학교 김재완 교수



<제목 차례>

1. 자율 주행기술의 개요	1
1.1. 자율 주행 기술의 역사	1
1.2. 자율 주행차의 정의	2
1.3. 자율 주행차의 구성 요소	4
2. 자율 주행을 위한 위치 추정	9
2.1. GNSS를 이용한 위치 추정	9
2.2. HD 지도와 라이다를 이용한 위치 추정	15
2.3. 시각적 주행거리계	28
2.4. 센서 융합	33
3. 자율 주행을 위한 인지	39
3.1. 개요	39
3.2. 데이터 세트	39
3.3. 객체 탐지	43
3.4. 분할	50
3.5. 다중 센서 데이터 융합	52
3.6. 객체 추적	55
3.7. 결론	58
4. 예측 및 경로 계획	59
4.1. 개요	59
4.2. 트래픽 예측	63
4.3. 경로 계획(routing)	68
4.4. 결론	77
5. 결정, 계획, 제어	79
5.1. 동작 결정	79
5.2. 모션 계획(Motion Planning)	83
5.3. 피드백 제어	89
5.4. 결론	94
6. V2X 통신	95
6.1. 개요	95
6.2. V2X 표준	95
6.3. V2X 기술의 발전	97
6.4. V2X를 이용한 자율 주행 기술	99
6.5. 백엔드 시스템	106

7. 결론	110
7.1. 사회적 전망	110
7.2. 과제	110
참고 문헌	112

<그림 차례>

그림 1	3
그림 2	4
그림 2.1	12
그림 2.2	13
그림 2.3	15
그림 2.4	16
그림 2.5	17
그림 2.6	18
그림 2.7	21
그림 2.8	24
그림 2.9	26
그림 2.10	28
그림 2.11	33
그림 2.12	35
그림 2.13	36
그림 3.1	40
그림 3.2	41
그림 3.3	41
그림 3.4	42
그림 3.5	42
그림 3.6	42
그림 3.7	44
그림 3.8	44
그림 3.9	46
그림 3.10	46
그림 3.11	47
그림 3.12	48
그림 3.13	49
그림 3.14	49
그림 3.15	50
그림 3.16	51
그림 3.17	51
그림 3.18	53
그림 3.19	54
그림 3.20	55
그림 3.21	56

그림 3.22	58
그림 4.1	59
그림 4.2	64
그림 4.3	69
그림 4.4	73
그림 5.1	87
그림 5.2	89
그림 5.3	91
그림 5.4	93
그림 6.1	96
그림 6.2	96
그림 6.3	100
그림 6.4	100
그림 6.5	101
그림 6.6	102
그림 6.7	102
그림 6.8	103
그림 6.9	104
그림 6.10	108
그림 6.11	108



1. 자율 주행기술의 개요

미국 도로교통안전국(NHTSA)의 최근 기술 보고서에 따르면, 도로 사고의 94%가 인간의 실수로 인해 발생한다[1]. 이러한 배경에서, 자율 주행 기술은 사고 방지, 배출 감소, 거동이 불편한 사람의 수송 및 운전 관련 스트레스 감소라는 미래를 내세우며, 개발되고 있다[2]. 자율 주행 자동차의 대중화가 실현된다면, 교통 혼잡 완화, 도로 사고 감소, 에너지 소비 감소, 운전 시간 재할당으로 인한 생산성 향상을 통해 자율 주행 기술로 인한 연간 사회적 편익이 2050년까지 거의 8,000억 달러에 이를 것으로 예상된다[3].

차량 역학에 대한 축적된 지식, 딥러닝의 출현으로 인한 컴퓨터 비전 기술의 비약적인 발전, 그리고 라이다와 같은 새로운 센서의 사용은 자율주행 기술의 발전을 촉진했다. 또한, 대중의 관심과 시장 잠재력의 증대는 다양한 수준의 자동화와 함께 자율 주행 기술의 출현을 촉발시켰다. 그러나 미성숙한 시스템으로 인한 사고는 신뢰를 훼손하고 나아가 생명을 앗아간다[4-6]. 그런 만큼 자율 주행 기술에 대한 이해결 과제와 첨단기술에 대한 철저한 조사가 필요하다고 판단된다.

1.1. 자율 주행 기술의 역사

자율주행차는 1980년대 중반부터 전 세계의 많은 대학, 연구 센터, 자동차 회사, 그리고 다른 산업의 회사들에 의해 연구되고 개발되었다. 최근 20년간 자율주행차 연구 플랫폼의 중요한 예로는 Navlab의 모바일 플랫폼[7], 파비아 대학과 파르마 대학의 자동차, ARGO[8], 그리고 UBM의 차량인 VaMoRs와 VaMP[9]가 있다.

Eureka Project PROMTHEUS [10]는 1987년에서 1995년 사이에 유럽에서 수행되었으며, 가장 초기의 주요 자동 운전 연구 중 하나였다. 이 프로젝트는 다임러-벤츠의 VITA II 개발로 이어졌고, 고속도로에서 자동 주행에 성공했다[11]. 2004년 미국 국방부가 주관한 DARPA(Defense Advanced Research Projects Agency) 그랜드 챌린지는 참가자 전원이 150마일 오프로드 주차장을 완주하지 못한 최초의 주요 자동운전 대회였다. 도전의 어려움은 결승전이 진행되는 동안 어떤 수준의 인간 개입도 허용되지 않는다는 규칙이었다. 또 다른 유사한 DARPA 그랜드 챌린지가 2005년에 개최되었다. 이번에는 다섯 팀이 인간의 간섭 없이 오프로드 트랙을 완주했습니다[12].

도시 장면에서 완전 자동화된 운전은 초기 시도 이후 이 분야의 가장 큰 도전으로 간주 되었다. 2007년에 개최된 DARPA Urban Challenge [13] 동안, 전 세계의 많은 다른 연구 단체들은 전형적인 도시 장면을 모델로 한 테스트 환경에서 ADS를 시도 했습니다. 6개 팀이 간신히 그 경기를 마쳤다. 이 대회가 당시까지



가장 크고 중요한 행사였음에도 불구하고, 테스트 환경은 보행자와 자전거 이용자와 같은 실제 도시 주행 장면의 특정 측면이 부족했다. 그럼에도 불구하고, 6개 팀이 도전을 완주했다는 사실은 상당한 관심을 끌었다. DARPA Urban Challenge 이후, 여러 국가에서 [14-17]]과 같은 몇 가지 더 많은 자동 주행 대회가 열렸다.

이를 계기로 해서 자동차 소프트웨어와 로봇틱스 기술이 엄청나게 발전했으며, SDV(Software Defined Vehicle) 분야 기술이 발전하는 데 전환점이 됐다. 그때부터 BMW, 아우디, 벤츠, 구글, 테슬라, 우버, 바이두를 비롯한 여러 회사가 자율주행차 관련 기술을 다양한 방식으로 개발하고 있다.

한편 세계 각국의 정책 입안자는 미래의 자율 주행차 시대를 대비해 규제안을 마련하기 시작했다. 보험과 표준 기반 시설과 관련 기술에 이르기까지 자동차 생태계 전체가 새로운 도전을 맞이하기 위해 적극적으로 활동하고 있다.

1.2. 자율 주행차의 정의

자율 주행차에 대한 생각은 사람마다 다양각색이다. 운전자가 전혀 필요 없는, 완벽히 자동으로 움직이는 자동차라고 생각하는 사람도 있고, 스스로 결정할 줄은 알지만, 비상시에는 사람이 운전대를 잡을 준비를 해야 한다고 생각하는 사람도 있다.

스스로 움직이는 자동차란 기술이 상당히 복잡하다고 생각할 수 있지만, 핵심 개념만 보면 상당히 단순하며 현재 수준의 기술로도 충분히 실현할 수 있다. 가령 자동차를 운전하는 과정을 생각해보자. 먼저 목적지부터 설정하고, 출발지로부터 이동하는 경로를 결정한 후에 출발한다. 운전하는 동안 시선은 항상 주변 상황을 주시한다. 주변 상황에는 건물, 가로수, 표지판, 도로변에 주차된 차와 같은 정적인 대상도 있고, 보행자, 주행 중인 차, 동물과 같은 동적인 대상도 있다. 달리다 보면 이런 대상이 도로 앞에 나타날 수도 있는데 이에 따라 적절히 대응해야 한다. 운전자는 자동차에서 제공하는 모든 수단을 활용해 원하는 방향으로 차를 이동 시켜야 한다. 이 과정에서 운행에 관련된 모든 규칙을 반드시 준수 해야한다.

이렇게 표현하면 상당 부분을 사람의 개입 없이 자동화할 수 있겠다는 생각이 든다. 실제로 일정 수준 이상으로 자동화된 차들이 많이 나와 있다. 비행기나 기차, 배도 어느 정도 자율 주행이 가능하다. 현재 수준의 컴퓨팅 파워에 각종 센서와 똑똑한 알고리즘과 다양한 구성 요소를 조합하면 운전자를 충분히 흉내 낼 수 있다.

자율 주행 기술의 수준과 기준을 업계에 명확히 제시하기 위해 미국 교통부(US DOT, US Department of Transportation)의 도로교통안전국(NHTSA, National Highway Traffic Safety Administration)이나 독일 연방도로연구소(German Federal



Highway Research Institute)와 같은 기관에서는 자율 주행의 수준에 따라 자율주행차를 정의했다. 그중에서도 국제자동차기술자협회(SAE International Automotive Engineers International, 구 미국자동차공학회)에서 운전 자동화에 대해 분류한 J3016[19]이 가장 널리 인용된다. 이 기준(그림 1)에 따르면, 자율 주행차를 비자동화(수동)부터 완전 자동화까지 여섯 단계로 구분한다.

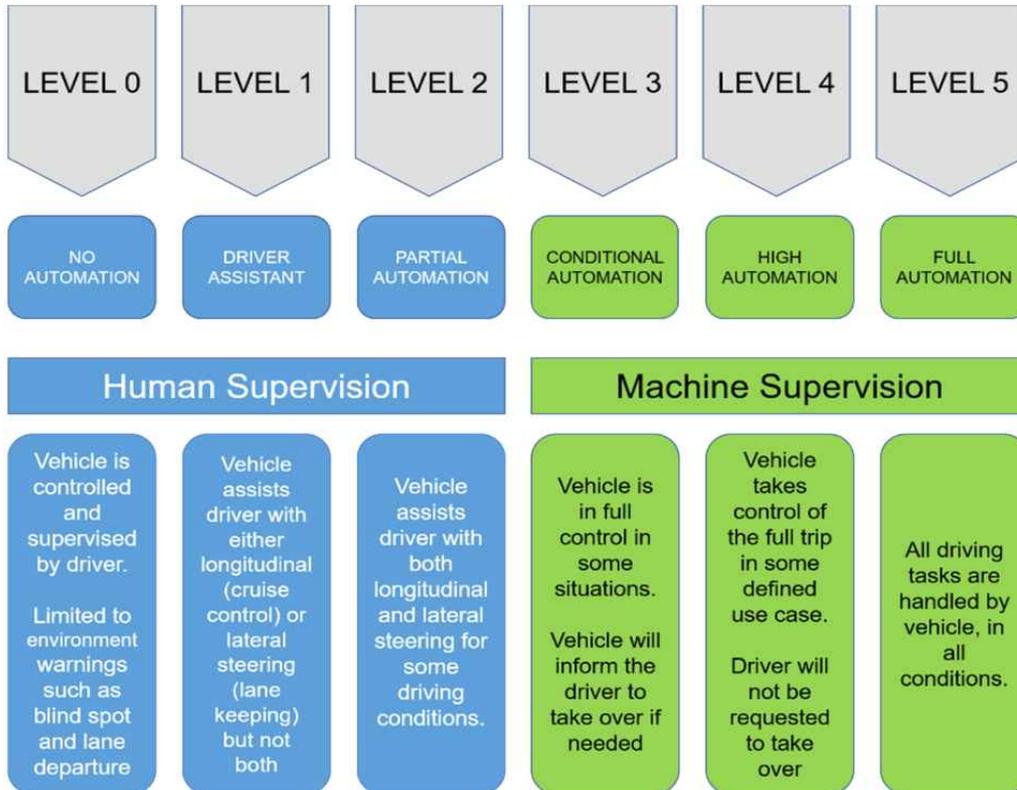


그림 1. SAE에서 기술한 자율주행차의 단계 구분(출처: [18])

가장 낮은 단계는 레벨 0으로서 운전과 관련된 조직과 주변 환경을 모니터링하는 활동을 모두 사람 운전자가 수행하며, 자동차 시스템은 경고 수준으로만 개입한다. 레벨 1은 운전자 보조 기능을 추가해 시스템이 횡방향 제어(lateral control) 또는 조향(steering)과 종방향 제어(longitudinal control) 또는 가속(acceleration)과 감속(deceleration)을 담당할 수 있다. 자동차를 시스템이 제어하지만, 여전히 사람 운전자가 제어권을 갖고 자동차 주변 환경을 지속적으로 모니터링해야 하며 필요한 시점에 언제든지 수동으로 개입해야 한다.

레벨 2 (부분 자율주행)는 자동차가 스스로 주행하는 첫 단계로서 일정하게 정의된 범위에서 횡방향 제어와 종방향 제어를 시스템이 담당한다. 이 레벨 역시 운전자가 차량과 주변 환경을 지속적으로 모니터링하면서 언제든지 수동 조작을 할 수 있도록 준비하고 있어야 한다. 레벨 3 (조건부 자율주행)부터 수준이 크게 달라진다. 시스템은 횡방향 및 종방향 제어뿐만 아니라, 한계 상황을 인지해 운전자

에게 알려준다. 운전자는 상황을 모니터링할 필요가 없지만, 시스템이 요청하는 시점에 언제든지 제어권을 가져올 수 있도록 준비하고 있어야 한다.

레벨 4(고도 자율주행)는 일정하게 정의된 범위의 모든 동작을 시스템이 담당하며, 운전자는 모니터링하거나 대기할 필요가 없다. 마지막 단계인 레벨 5(완전 자율주행)는 운전과 관련된 모든 범위의 동적 운전 작업을 시스템이 전적으로 책임진다. 이 말은 출발지부터 목적지까지 사람의 개입없이 자동차를 완전히 제어한다는 것을 의미한다.

1.3. 자율 주행차의 구성 요소

그림 2는 자율주행 시스템의 전통적인 구조를 보여준다. 그림 2를 보면 자율 주행(autonomous driving)이란 하나의 기술이 아닌, 여러 가지 서브 시스템이 상당히 복잡하게 구성된 시스템이다. 서브시스템은 크게 세 가지로 나뉜다. 첫째는 센싱(sensing), 인지(perception), (복잡한 상황을 추론하기 위한) 의사결정(decision)을 처리하는 알고리즘이고, 둘째는 운영체제와 하드웨어 플랫폼으로 구성된 클라이언트 시스템이고, 셋째는 HD(high-definition) 지도와 딥러닝 모델 학습(deep learning model training), 시뮬레이션, 데이터 스토리지 등을 제공하는 클라우드 플랫폼이다.

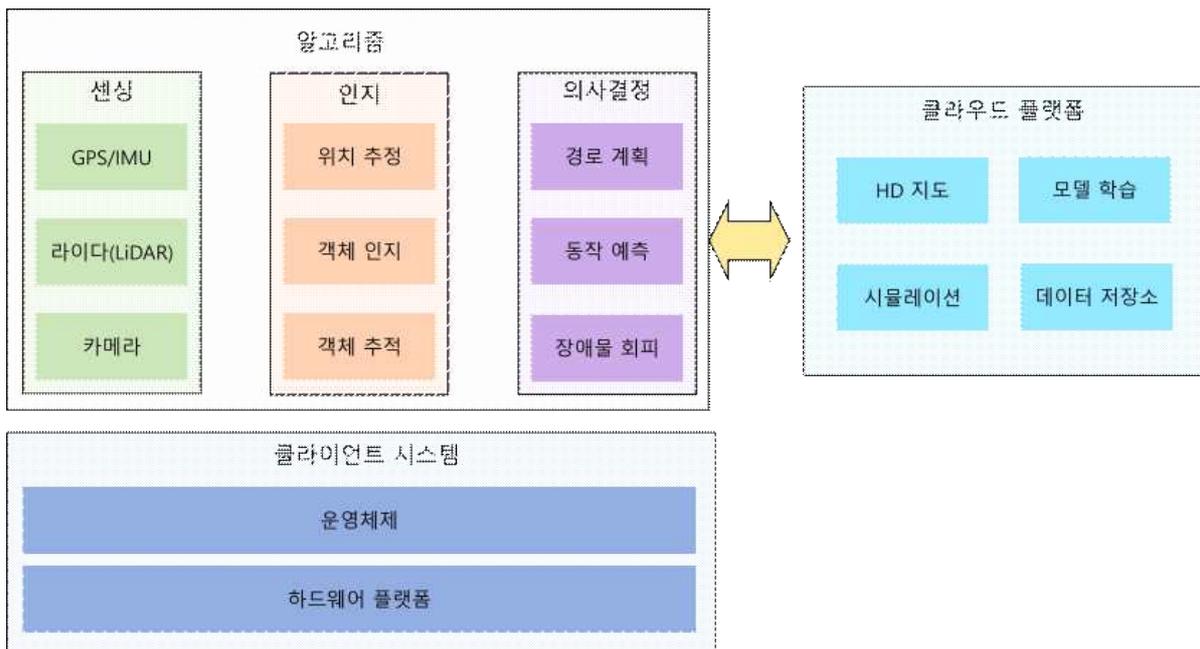


그림 2. 자율 주행 시스템의 구조

첫 번째 서브시스템인 알고리즘은 센서로 수집한 원본 미가공(raw) 데이터로부터 의미 있는 정보를 추출해 주변 환경을 파악하고 다음 동작을 결정한다. 두 번



째 서브시스템인 클라이언트 시스템은 실시간 및 신뢰성 요구사항을 만족하도록 여러 알고리즘을 서로 엮는다. 예를 들어 카메라에서 60Hz 주기로 생성하는 데이터를 제때 처리하려면, 클라이언트 시스템의 프로세싱 파이프라인에서 가장 긴 작업도 최대 16ms 이내에 작업을 완료해야 한다. 세 번째 서브시스템인 클라우드 플랫폼은 자율 주행차에 필요한 오프라인 컴퓨팅 및 스토리지 기능을 제공한다. 클라우드 플랫폼을 활용하면 새로운 알고리즘을 테스트하고 HD 지도를 업데이트하고, 객체 인지와 추적 및 의사결정 모델에 대한 트레이닝 수준을 높일 수 있다.

1.3.1. 센서

자율주행에 사용되는 일반적인 센서로는 범지구 위성항법 시스템(Global Navigation Satellite System: GNSS), 라이다(Light Detection and Ranging: LiDAR), 카메라, 레이더 및 소나가 있다. GNSS 수신기 특히 실시간 이동위치추정(RTK) 기능이 있는 수신기는 최소한 미터 수준의 정확도로 글로벌 위치를 업데이트하여 자율주행차가 스스로 위치를 파악하는데 도움을 준다.

- 라이다(LiDAR): 라이다는 일반적으로 HD 지도생성, 실시간 위치 추정 및 장애물 회피에 사용된다. 라이다는 거리 계산을 위해 레이저 빔의 표면산란과 반사시간 측정을 통해 작동한다. 라이다 장치에는 두 가지 문제점이 있다. 첫째는 라이다는 매우 비싸다는 것이고 (자율주행 등급용 라이다의 비용은 8만 달러를 넘을 수 있음), 둘째는 폭우나 짙은 안개와 같은 악천후 조건에서는 정확한 측정을 제공하지 못할 수 있다는 것이다.

- 카메라(camera): 카메라는 주로 차선 검출 신호등 검출 및 보행자 검출과 같은 물체 인식 및 추적 작업에 사용된다. 기존 방식에서는 대부분 차량 주변에 여러 대의 카메라를 장착하여 물체를 검출하고 인식과 추적을 수행한다. 그러나 카메라 센서의 중대한 문제는 악천후 조건에서 제공하는 데이터가 신뢰하지 못할 수 있다. 그리고 카메라에서 제공된 엄청난 양의 데이터는 높은 컴퓨팅을 요구한다는 것이다. 이러한 카메라는 일반적으로 60Hz에서 실행되는 것을 고려하면 초당 1GB 이상의 데이터가 생성될 수 있음을 유의해야 한다.

- 레이더(radar)와 소나(sonar): 레이더와 소나(초음파 탐지) 시스템은 장애물 회피의 최후의 보루로써 사용된다. 레이더와 소나에 의해 생성된 데이터는 차량 경로 앞의 가장 가까운 물체와의 거리를 보여준다. 레이더의 주요 장점은 모든 기상 조건에서 작동한다는 것이다. 소나는 일반적으로 0~10m 거리의 물체를 검출하는 반면 레이더는 3~150m 거리의 물체를 검출한다. 결합된 이들 센서의 가격은 5000 달러 미만이다.



1.3.2. HD 지도

전통적인 디지털 지도는 일반적으로 위성사진으로부터 생성되고 미터 수준의 정확도를 가진다. 이 정확도는 사람 운전자에게는 충분하지만, 자율주행차량은 차선 수준의 정보를 위해 더 높은 정확도를 필요로 한다. 따라서 자율주행을 위해서는 HD 지도가 필요하다.

기존 디지털 지도와 마찬가지로 HD 지도에는 많은 정보 계층이 있다. 하단 계층에서는 위성 이미지를 사용하는 대신 라이다 가공 전 원래의 데이터로 5cm x 5cm의 그리드 단위로 그리드 지도가 생성된다. 이 그리드는 기본적으로 각 셀의 주변 환경에 대한 고도 및 반사 정보를 기록한다. 자율주행차가 이동하고 새로운 라이다 스캔을 수집함에 따라 자율 주행차는 새로운 라이다 스캔 및 그리드 지도와 함께 GNSS에서 제공되는 초기 위치 추정치를 실시간 비교하여 자체 위치추정을 수행한다[7].

그리드 계층 위에는 여러 계층의 시멘틱 정보가 있다. 예를 들어 차선정보가 그리드 지도에 추가되어 자율 주행차가 이동할 때 올바른 차선에 있는지 확인할 수 있다. 차선정보에는 교통표지 라벨이 추가되어 자율 주행차의 로컬속도 제한, 신호 등이 근처에 있는지 여부 등을 알려준다. 이는 자율 주행차의 센서가 신호를 포착하지 못하는 경우를 대비하여 추가 보호계층을 제공한다.

기존 디지털 지도의 재생사이클은 6~12개월이다. 그러나 HD 지도에 최신 정보가 포함되는지 확인하기 위해서는 HD 지도의 재생사이클을 일주일 이내로 줄여야 한다. 결과적으로 중규모 도시에 대한 HD 지도의 운영과 생성 그리고 유지 관리에만 연간 수백만 달러 이상의 비용이 들 수도 있다.

1.3.3. 컴퓨팅 시스템

경로계획 및 제어 알고리즘과 시물인식 및 추적 알고리즘은 각각 다른 종류의 프로세서를 요구하는 매우 상이한 행동특성을 가지고 있다. 반면에 HD 지도는 메모리의 필요성이 강조된다[8]. 따라서 제한된 컴퓨팅 리소스와 전력의 수요예측 안에서 이러한 요구를 모두 해결하는 컴퓨팅 하드웨어 시스템을 설계하는 것이 필수적이다. 예를 들면 문헌 [8]과 같이, 자율주행 컴퓨팅 시스템의 초기 설계에는 Intel Xeon E5 프로세서, PCI-E(Peripheral Component Interconnect-E) 버스와 연결된 4~8개의 Nvidia K80 GPU(graphics processing unit) 가속기가 장착되었다. 64.5 TOPS(Tera Operations Per Second)를 제공할 수 있는 전체 시스템은 최대 약 3000W를 소비하여 엄청난 양의 열을 발생시킨다. 또한 전체 솔루션의 비용은 3만 달러에 이르러 일반 소비자들이 구매하고 수용하기에는 어려움이 있다.



1.3.4. 센서 융합

위치추정 또는 인식을 위해 라이다만 사용하는 것은 높은 비용을 필요로 하는 반면, 검출 결과를 신뢰할 수 없을 수도 있다. 경제성과 신뢰성을 달성하기 위해서는 여러 개의 저렴한 센서(카메라, GNSS 수신기, 휠 인코더, 레이더 및 소나)를 사용하고 데이터를 융합하여 시너지 효과를 낼 수 있다. 이러한 센서는 각각 고유의 특성, 단점 및 장점을 가지고 있을 뿐만 아니라 하나가 고장나거나 오작동하는 경우 다른 센서가 즉시 인계하여 시스템 신뢰성을 보장할 수 있도록 서로를 보완한다. 이러한 센서 융합 접근 방식의 비용은 2천달러를 넘지 않는다.

위치추정 서브시스템은 GNSS 수신기에 의존하여 서브미터 수준의 정확도로 초기 위치추정을 제공한다. 비주얼 주행거리계(visual odometry)는 위치추정 정확도를 데시미터 수준까지 더욱 향상시킬 수 있다. 또한 휠 인코더를 사용하는 경우 GNSS 수신기 및 카메라가 고장나더라도 차량의 움직임을 추적할 수 있다. 비주얼 주행거리계는 두 프레임 간 중첩을 분석하여 위치 변화를 추론하는데 급회전과 같은 갑작스러운 움직임이 차량에 가해지면 두 연속 프레임 사이에 중첩 영역이 부족하여 위치추정을 유지하지 못할 가능성이 있다.

능동 인식 서브시스템은 차량이 주변 환경을 인식할 수 있도록 지원한다. 컴퓨터비전과 밀리미터파(mmWave) 레이더의 이해와 조합을 바탕으로 50m 거리 내의 정적, 동적 물체 검출과 추적이 가능하며 이를 통해 차량은 원활하고 안전한 주행을 확보하기 위한 행동을 결정할 수 있다. 스테레오 비전을 사용하면 보행자와 움직이는 차량을 포함한 물체를 쉽게 인식할 수 있을 뿐만 아니라 검출된 물체까지의 거리도 정확하게 파악할 수 있다. 또한, mmWave 레이더는 어떠한 기상 조건에서도 빠르게 움직이는 물체와 거리를 검출하고 추적할 수 있다.

수동 인식 서브시스템은 즉각적인 위험을 검출하는 것을 목표로 하며 차량의 마지막 방어선 역할을 한다. 차량주변 0~5m 거리를 다루며 mmWave 레이더와 소나의 조합으로 이루어진다. 레이더는 매우 우수한 동적 물체 검출기이고 소나는 매우 우수한 정적 물체 검출기이다. 근거리에서 무언가가 검출되면 차량의 안전을 확보하기 위한 다양한 방법들이 현재 차량 속도에 따라서 수립된다.

1.3.5. 컴퓨터비전

컴퓨터 비전은 위치추정과 능동 인식에 모두 활용될 수 있다. 위치추정은 VSLAM (Visual Simultaneous Localization and Mapping) 기술에 의존하여 실시간으로 정확한 차량 위치를 얻을 수 있다. 그러나 VSLAM은 차량이 이동하는 거리가 길어질수록 누적 오차로 인해 위치 에러가 증가하는 현상이 발생한다. 다행히도 VSLAM과 GNSS 위치추정을 융합하면 다양한 조건에서 높은 정확도를 얻을 수 있



으며 이는 GNSS가 차단되지 않은 경우 집단진실(group-truth) 데이터로 사용될 수 있고 GNSS가 차단된 경우에는 VSLAM이 높은 정확도를 제공할 수 있기 때문이다.

또한 컴퓨터비전은 능동 인식에도 사용될 수 있다. 스테레오 비전을 사용하여 다양한 객체의 공간 또는 깊이 정보를 추출할 수 있고, 심화학습(deep learning) 기법을 사용하여 다양한 객체의 시멘틱 정보를 추출할 수 있다. 공간, 시멘틱 정보들을 융합하면 보행자 및 자동차 뿐만 아니라 이들의 현재 차량과의 거리와 같은 관심 대상들을 파악할 수 있다.

1.3.6. 경로계획 및 제어

경로계획 및 제어 모듈은 객체 인식 및 위치 추정 모듈로부터 입력을 받고 실시간으로 결정을 내린다. 일반적으로 서로 다른 조건에 따라 경로계획 및 제어 모듈은 다른 행동들을 정의하며 이 중 하나의 행동이 선택된다.

일반적인 경로계획 및 제어 시스템의 구조는 다음과 같다. 먼저 사용자가 목적지를 입력하면 라우팅 모듈이 지도에서 도로 네트워크 정보를 확인하고 경로를 생성한다. 그리고 경로가 동작 계획 모듈에 공급되면 트래픽 규칙을 확인하여 움직임 규격을 생성한다. 다음으로 움직임 규격과 함께 생성된 경로는 동작 플래너(planner)로 전달되어 실시간 인식 및 위치 정보를 결합하여 궤적을 생성한다. 최종적으로 생성된 궤적은 제어 시스템으로 전달되어 계획된 움직임의 실행 오류를 정정 한다.



2. 자율 주행을 위한 위치 추정

2.1. GNSS를 이용한 위치 추정

사람이 운전할 때는 주로 위성항법시스템(GNSS, global navigation satellite system)을 활용해 위치 추정을 수행한다. 자율 주행 자동차의 위치 추정 작업도 GNSS로 처리할 수 있다. 이 절에서는 GNSS와 관련된 기술을 자세히 소개하고, 자율 주행에 GNSS를 적용할 때의 장단점을 살펴 본다.

2.1.1. GNSS 개요

GNSS에는 GPS, GLONASS, Galileo, BeiDou 등이 있다 GNSS의 개념을 이해하기 쉽도록 GPS를 예로 들어 살펴보자. GPS는 수신기가 처리할 수 있는 형태로 부호화된 위성 신호를 보낸다. GPS 수신기는 이 신호를 받아서 위치와 속도와 시간을 추정한다[20]. 이 작업을 제대로 처리하려면 4개의 GPS 신호를 받아서 3차원상의 위치와 수신기 클럭 기준으로 발생한 시간차를 계산해야 한다. GPS 위성은 지표면으로부터 약 20200km 높이에 있는 원형에 가까운 6개의 궤도에 분산 배치된다. 각 궤도는 적도를 기준으로 55° 기울어져 있으며 지구 한 바퀴를 도는 데 약 11시간 58분이 걸린다.

위성이 송신하는 신호는 $f_0 = 10.23\text{MHz}$ 의 기본 주파수를 갖도록 생성한다[20]. 이 때 신호는 원자 시계를 기준으로 정확하게 동기화되는데 오차는 하루에 약 10^{-13} 초에 불과하다. L-band의 두 반송파 신호 L1 및 L2는 f_0 의 정수 곱셈 연산에 의해 생성된다. 반송파 L1 및 L2는 코드에 의해 2상 변조되어 수신기에 위성 시각 값을 제공하고 궤도 파라미터와 같은 정보를 전송한다. 코드는 +1 또는 -1 상태의 시퀀스로 구성되며 이들은 이진 값 0 또는 1에 해당한다. 2상 변조는 코드 상태가 변경될 때마다 반송파 위상을 180° 변경하여 수행된다. 위성신호에는 위성궤도, 궤도 섭동, GPS 시간, 위성시각, 전리층 파라미터 및 시스템 상태 메시지 등에 대한 정보가 포함된다. 내비게이션 메시지는 각 프레임 별 1500비트를 포함하는 25개의 프레임으로 구성되며 각 프레임은 300비트의 5개의 서브 프레임으로 세분화된다.

GNSS에서 두 번째로 중요한 부분은 기준 좌표계의 정의다. 기준 좌표계(reference coordinate system)는 위성의 움직임을 표현하고, 관측 대상을 모델링하고, 측정 결과를 해석하는 데 핵심적인 역할을 한다 GNSS가 제대로 작동하려면 2개의 기준 좌표계가 필요하다. 하나는 위성의 움직임을 표현하는 공간 고정 관성 좌표계(space-fixed, inertial reference system)이고, 다른 하나는 관측 위치와 위성



측지(satellite geodesy)의 결과를 표현하는 지구 고정 지형 기준 좌표계(earth-fixed, terrestrial reference system)이다. 이렇게 두 좌표계를 사용할 때 지구고정 좌표계와 공간 고정 좌표계 사이의 좌표 변환 매개변수를 주로 사용하는데, GNSS 수신기나 후처리 소프트웨어는 지구 고정 좌표계 기준으로 수신기 위치를 계산하는 데 이 값을 직접 활용한다. 지구 기준 좌표계는 서로 직교하는 세 축으로 정의하며 지구의 자전축에 정렬시킨 Z축을 CIO(Conventional International Origin)라 정의한다. X축은 그리니치 자오선을 향하고, Y축은 X축과 Z축에 수직을 이루면서 오른손 좌표계(right-handed coordinate system)를 형성한다. GPS는 지심 등위 회전 타원체를 표현하는 WGS84로 기준 좌표계를 정의하고 있다[21].

최근 몇 년 동안 여러 성상도를 지원하는 GNSS 수신기의 출현은 상공의 GNSS 위성 수가 증가함에 따라 꾸준한 속도를 유지하고 있다. 새로운 GNSS 성상도 발전으로 대부분의 새로 구매한 장치는 다중 성상도를 지원할 것으로 예상된다. 다중 성상도 지원의 이점으로는 음영 영역에서의 가용성 증가 그리고 더 많은 위성의 시야에 따른 높은 정확도 및 독립 시스템의 스푸핑의 어려움에 따른 강인성 향상이 있다.

이론적으로는 GNSS를 사용하면 오류없이 완벽한 위치추정 결과를 얻을 수 있다. 그러나 GNSS에서 오류가 발생할 수 있는 부분이 여러 곳 존재하며 오류가 발생할 수 있는 잠재적 원인은 다음과 같다.

- 위성 시계 오차(satellite clock error): GNSS 위성 내부의 원자 시계에 조금이라도 오차가 발생하면 수신기에서 계산한 위치에 오차가 크게 발생한다. 대략 10ns정도의 시계 오차로 3m가량의 위치 오차가 발생한다.

- 궤도 오차(orbit error): GNSS 위성은 미리 정의한 궤도를 따라 정밀하게 움직인다. 하지만 위성 시계와 마찬가지로 궤도에도 약간씩 오차가 발생한다. 위성의 궤도에 오차가 발생하면 지상의 컨트롤 시스템은 이를 보정하는 신호를 보내서 위성의 예상 위치를 나타내는 위성 궤도력(satellite ephemeris)을 업데이트한다. 이렇게 지상 컨트롤 시스템을 통해 교정해도 미세한 궤도 오차는 여전히 남아 있는데 이로 인해 최대 2.5m의 위치 오차가 발생할 수 있다.

- 이온층 지연(ionospheric delay): 이온층은 대기층에서 지구 표면으로부터 80~600km 떨어진 영역으로서 이온(ion)이라고 불리는 전기적 성질을 띠는 입자로 이뤄져 있다. 이러한 이온의 영향으로 인해 위성의 신호가 지연되는 현상이 발생해 대략 5m 범위로 상당히 큰 위치 오차가 발생한다. 이온층 지연의 정도는 태양의 활동, 연도, 계절, 시간, 위치 등에 따라 달라진다. 그래서 이온층 지연이 위치 오차에 미치는 영향을 정확히 파악하기 힘들다. 또한 이온층을 통과하는 인공위성 신호의 주파수에 따라 이온층 지연 시간이 달라진다.

- 대류층 지연(tropospheric delay): 대류층은 지구 표면과 가까운 대기층이다. 대



류층 지연의 정도는 대류층 내부의 습도 온도 대기압에 따라 달라진다. 대류층의 환경은 지표에서의 환경과 상당히 비슷하기 때문에 기지국과 수신기에서 발생하는 대류층 지연은 상당히 비슷하다. 그래서 RTK GNSS로 대류층 지연을 보정한다. RTK에 대해서는 다음 절에서 설명한다.

- 다중경로오차(multipath error): 다중 경로 오차란 빌딩의 벽과 같은 물체에 반사된 GNSS 신호를 안테나가 수신할 때 발생한다. 반사된 신호의 이동 거리가 더 길기 때문에 다른 신호보다 수신기에 더 늦게 입력된다. 이렇게 지연된 신호로 인해 수신기에서 계산한 위치에 오차가 발생한다.

2.1.3. 위성 기반 오차 보정 시스템

SBAS(Satellite-Based Augmentation Systems)는 GNSS의 측정 오차를 보정하는 역할을 한다. 특히, GNSS에서 부족한 정확성, 완전성, 지속성, 유효성 등을 SBAS로 보완할 수 있다. SBAS는 전 대륙에 걸쳐 정확한 위치에 배치된 기준국을 통해 GNSS 측정치를 얻는다. 여기서 발생한 GNSS 오차를 컴퓨팅 센터로 보내서 차분 보정 값을 계산하고 무결성(integrity) 메시지를 만들어서 정지 궤도 위성을 통해 기존의 GNSS 메시지에 덧붙이거나 덮어씌우는 식으로 대륙 전체로 브로드캐스트한다. SBAS 메시지는 정지 궤도 위성을 이용해 브로드캐스트하기 때문에 광범위하게 안정적으로 전달할 수 있다.

여러 나라에서 SBAS 시스템을 직접 구축했다. 유럽은 EGNOS(European Geostationary Navigation Overlay Service)를 보유하고 있으며 주로 유럽 대륙을 대상으로 제공하고, 미국은 WAAS(Wide Area Augmentation System)를 보유하고 있다. 중국도 자체적으로 구현한 SBAS를 제공하는 BDS(BeiDou System)를 개발했다. 일본은 MSAS(Multi-functional Satellite Augmentation System)라는 시스템을 보유하고 있으며, 인도는 GAGAN(GPS and GEO Augmented Navigation)이란 이름으로 자체 개발한 SBAS를 인도 전역에 제공하고 있다. 이렇게 각 나라마다 개발한 SBAS 시스템은 세계 공통 표준을 따르기 때문에 서로 연동할 수 있다.

참고로 상용 GNSS 수신기는 대부분 SBAS 기능을 제공한다. 특히 WAAS는 최소 95%의 측정 시간 동안 수평축과 수직축에 대한 위치 정확성이 7.6m 이하여야 한다고 규격에 명시하고 있다. 실제로 WAAS에서 측정한 결과를 보면, 대부분의 미국 영토에서 수평축과 수직축에 대해 각각 1.0m, 1.5m 이하의 정확성을 보이는 것을 관측할 수 있다.

2.1.4. RTK와 DGPS

일반적으로 다중 위성군을 지원하는 상용 GNSS 시스템의 위치 추정 정확성은

대부분 반경 2m 정도이다. 이 정도의 정확성은 사람이 자동차를 운전하는 데는 충분할지 몰라도, 자율 주행 자동차가 정상적으로 주행하는 데는 부족하다. 자율 주행 자동차가 차선을 유지하려면 위치 추정 정확성이 센티미터 수준이어야 한다. 이 정도의 정밀도를 확보하려면 RTK(Real-Time Kinematic)나 DGNS(Differential GNSS)를 적용해야 한다. 이 절에서는 RTK와 DGNS의 작동 방식을 소개한다.

RTK GNSS는 위성의 시계 오차, 궤도 오차, 이온층 지연 오차, 대류층 지연 오차를 줄이는 방식으로 정확성을 높인다. 그림 2.1은 RTK GNSS의 기본적인 개념을 보여주고 있다. 이러한 GNSS 오차를 보정하기 위한 좋은 방법은 위치를 정확히 알고 있는 기준국에 GNSS 수신기를 설치하는 것이다. 기준국 수신기는 위성 데이터를 통해 수신기의 위치를 계산해서 이 값을 현재 알고 있는 위치와 비교해 오차를 계산한다. 그 결과로 나온 오차 보정 정보를 기준국에서 자동차로 전달한다.

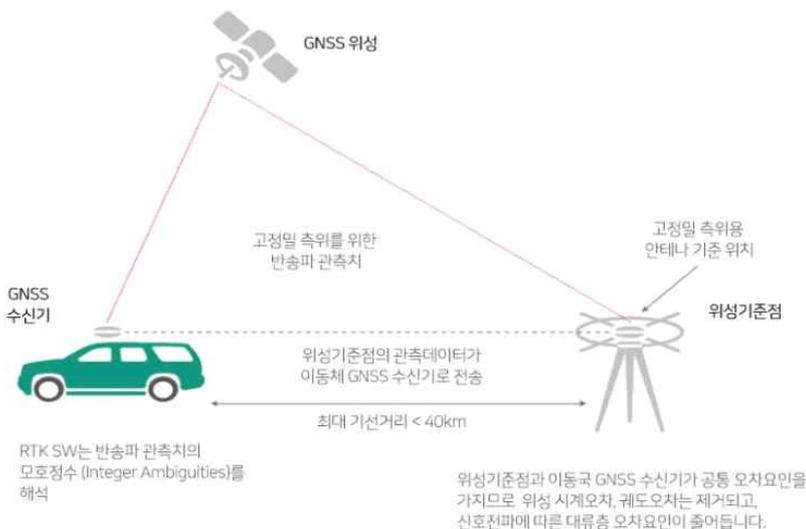


그림 2.1. RTK GNSS (출처: 서울시 GNSS 측위 서비스)

구체적으로 살펴보면 RTK는 반송파 기반 거리 측정을 사용하며 코드 기반 위치 측정 기술보다 훨씬 더 정확한 거리 및 위치추정을 제공한다. 코드 기반 위치 측정은 C/A(coarse acquisition) 코드 수신기를 통해 데이터를 수집하는 하나의 처리 기술이며 위성의 의사난수(pseudorandom) 코드에 포함된 정보를 사용하여 위치를 계산한다. 차동 보정을 진행한 이 처리 기술은 5m 정확도를 제공한다. 반송파 기반 거리 측정은 반송파 위상 수신기를 통해 데이터를 수집하는 또 다른 처리 기술이며 라디오 반송파 신호를 사용하여 위치를 측정한다. 반송파 신호는 의사난수 코드보다 훨씬 더 높은 주파수를 갖으며 의사난수 코드 하나만 사용하는 것보다 더 정확하다. 의사난수 코드는 기준을 좁히고 반송파 코드는 기준을 더욱 좁힌다. 차동 보정을 진행한 이 처리 기술은 서브미터 정확도를 제공한다. 반송파 기반 거



리 측정에서는 위성과 차량 사이의 반송파 주기 수를 결정한 다음 여기에 반송파 파장을 곱하여 거리를 측정한다. 측정된 거리에는 여전히 위성 시각, 천체력, 전리층 및 대류권 지연과 같은 소스의 오류가 포함된다. 이러한 오류를 제거하고 반송파 기반 측정의 높은 정밀도를 활용하려면 기지국에서 차량으로 전송된 측정값을 반영한 RTK 실행이 필요하다.

RTK GNSS는 모호성 추정 기법(ambiguity resolution)과 차분 보정을 통합한 알고리즘으로 자동차의 위치를 판단한다. RTK GNSS를 통해 자동차에 보장되는 위치 정확도는 기준국과의 거리 및 차분 보정의 정확도에 따라 달라진다. 차분 보정의 정확도는 기준국의 위치가 좋고 기준국에서 수신하는 위성 정보의 질이 높을수록 더 높아진다. 간섭이나 다중 경로 오차 같은 환경의 영향을 최소화하도록 기준국의 위치를 잘 정하는 것은 기준국과 자동차의 수신기나 안테나의 품질 못지않게 중요하다.

2.1.5. PPP 알고리즘

RTK GNSS를 적용하면 자율 주행에 필요한 밀리미터급의 정확도를 얻을 수 있지만, 이렇게 하려면 기준국을 직접 설치해야 하는데 이를 관리하는 비용이 엄청나다. 이 절에서는 정밀 단독 측위(Precise Point Positioning, PPP) GNSS 시스템으로 이러한 문제를 해결하는 방법을 소개한다[22, 23].

그림 2.2는 PPP GNSS의 작동 과정을 보여준다.

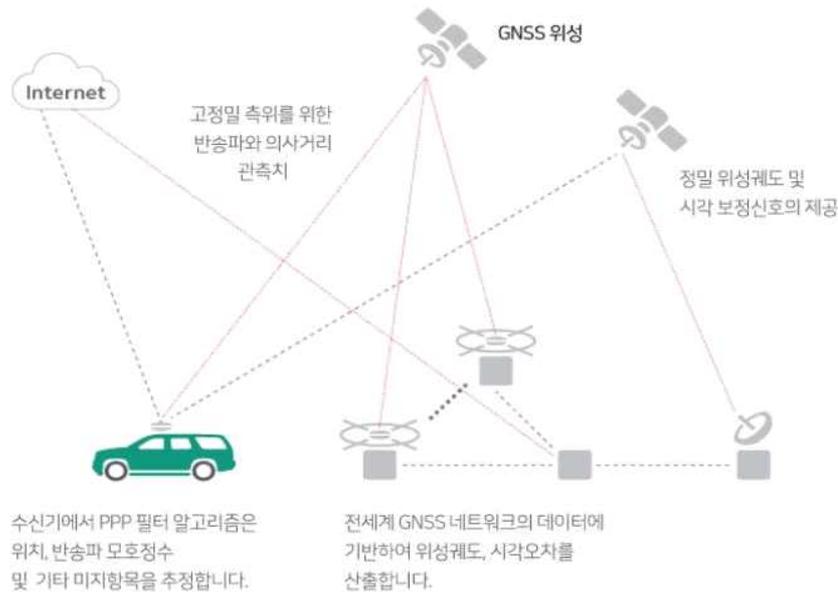


그림 2.2. PPP GNSS(출처: 서울시 GNSS 측위 서비스)

여러 기준국이 전 세계에 설치되어 있고, 각 기준국은 정밀한 기준 위성 궤도와



기준 GNSS 위성 시계 정보를 실시간으로 받아서 위성 위치 추정 결과에 적용할 보정 정보를 계산한다. 계산을 마친 보정 정보는 위성이나 인터넷을 통해 최종 사용자에게 전송된다. 위성의 위치와 시계 정보가 정밀할수록 위성 시계 오차와 궤도 오차를 최소화할 수 있다. 그런 다음, 이 중 주파수 GNSS 수신기를 통해 반송파 주파수에 비례하는 이온층 오차의 1차 효과를 제거한다. 이렇게 하면 이 중 주파수 GNSS 측정치를 통해 1차 이온층 지연을 완전히 제거할 수 있다. 또한 UNB 모델로 대류층 지연을 보정한다[24]. 정확도를 높이기 위해 위치나 기타 미지수를 추정할 때 잔여 대류층 지연도 측정한다[25]. PPP는 이렇게 여러 기법을 조합하기 때문에 센티미터 단위의 정확도로 위치를 알아낸다.

PPP는 하나 이상의 근접 기준국에서 관측 정보를 가져올 필요가 없고 RTK처럼 기준국에 대한 상대 위치가 아닌 절대 위치를 제공한다는 점에서 RTK와 다르다. PPP는 상대적으로 밀도가 낮은 기준국 네트워크에 있는 기준국 측정 정보로 지상 처리 센터에서 계산한 정밀한 궤도와 시계 데이터만 있으면 된다. 여기서 주목할 점은 PPP는 단 하나의 GPS 수신기만 사용하므로 사용자 주변에 기준국이 없어도 된다는 것이다. 이처럼 PPP는 글로벌 기준 프레임에 따른 위치 해를 계산하기 때문에 글로벌 위치 계산법이라고 볼 수 있다. 결론적으로 PPP는 위치 해의 성능이 로컬 기준국에 상대적으로 결정되는 RTK보다 위치 해의 일관성이 훨씬 뛰어나다. 또한 PPP는 SBAS 시스템과 구조가 비슷하다. SBAS에 비해 PPP의 주된 장점은 정밀한 GNSS 기준 궤도 정보와 시계 정보를 실시간으로 받아서 정확도를 센티미터급으로 높일 수 있다는 점이다. 이에 비해 SBAS는 미터 급의 정확도만 제공한다. 또한 PPP는 하나의 보정 스트림을 전 지역에서 사용할 수 있지만 SBAS는 특정 지역에서만 사용할 수 있다.

PPP에서 발생하는 가장 큰 문제는 대기 조건 위성의 배치 다중 경로 환경 같은 지역적인 편향을 보정해서 센티미터급의 정확도를 제공하는 데 약 30분가량의 상당히 긴 시간이 걸린다는 것이다. 통합 후처리 PPP 서비스는 현재 많이 나와 있지만, 실시간 PPP 시스템은 아직 시작 단계에 머물러 있다[26-28].

2.1.6. GNSS/INS 통합

지금까지는 그동안 발전한 GNSS 기술에 대해 살펴보았다. 이번 절에서는 관성 정보를 활용해 GNSS의 위치 추정 성능을 높이는 방법을 소개한다[29]. INS(Inertial Navigation System)는 IMU(Inertial Measurement)로 측정된 회전과 가속도 정보를 이용해 시간에 따른 상대적인 위치를 계산한다. 흔히 사용하는 6축 IMU는 3개의 직교축에 배치한 6개의 상호 보완(complimentary) 센서로 구성된다. 각 축마다 가속도계와 자이로스코프가 장치된다. 가속도계(accelerometer)는 선형 가속도를 측



정하고, 자이로스코프(gyroscope)는 회전 가속도를 측정한다. IMU는 이러한 센서를 이용해 3차원 공간의 상대적인 움직임을 정밀하게 측정한다. INS는 IMU로 측정된 값을 이용해 위치와 속도를 계산한다. 또한 IMU의 측정 값으로 각 축에 대한 각 속도 정보를 알 수 있는데, 이 값은 자세(롤, 피치, 방위각)를 추정하는 데 활용된다.

일반적으로 INS 시스템은 1kHz 수준의 매우 빠른 업데이트 주기로 위치 정보를 계산한다. 하지만 INS에는 몇 가지 단점이 있다. 우선 INS는 초기 시작점을 기준으로 상대적인 위치 해만 제공한다. 따라서 이러한 초기 시작점 정보를 반드시 제공해야 한다. 두 번째 문제는 더 심각하다. IMU를 통한 3차원 항법은 초당 수백에서 수천 번의 합 연산을 통해 이뤄지는데 이 과정에서 발생하는 측정 오차도 함께 누적된다. 다시 말해, 외부 기준을 이용해 보정하지 않으면 INS 시스템의 결과가 실제 위치와 급격히 멀어진다. 따라서 INS를 이용해 위치 추정을 수행할 때는 반드시 정확한 외부 기준 정보를 제공해야 한다. 그래야 칼만 필터 같은 수학적 필터로 위치 추정 오차를 최소화할 수 있다.

그림 2.3을 보면 GNSS로부터 외부 기준 정보를 얻는다. GNSS는 초기 시작점으로 사용할 절대 좌표를 제공한다. 또한 GNSS는 INS 필터 추정치를 업데이트하기 위한 연속적인 위치와 속도 값도 제공한다. 터널을 통과할 때와 같이 신호 방해로 인해 GNSS 신호가 손상되더라도 INS를 이용해 단기간의 자동차 위치를 알아낼 수 있다.

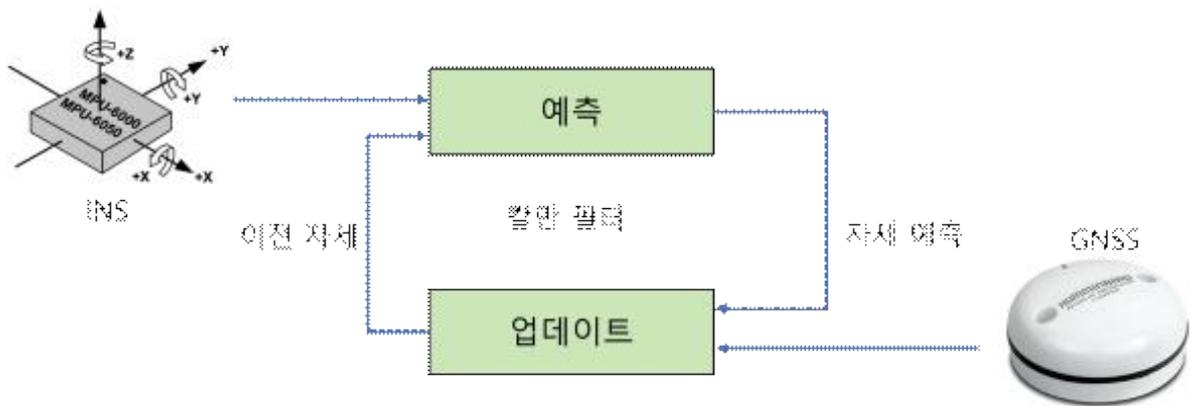


그림 2.3. GNSS/INS 융합

2.2. HD 지도와 라이다를 이용한 위치 추정

웨이모(Waymo), 바이두(Baidu), BMW 등에서 제작한 상용 자율 주행 자동차의 시제품은 대부분 라이다(LiDAR)와 HD 지도를 통해 위치 추정을 수행한다. 이 절에서는 라이다와 HD 지도의 원리를 알아보고 이 두 가지 장치를 조합해 자율 주행 자동차의 정확한 위치 추정을 수행하는 방법을 소개한다.

2.2.1. 라이다 개요

이 절에서는 라이다 기술을 개략적으로 소개한다. 라이다(LiDAR)란 ‘Light Detection And Ranging’의 약자로, 목표물에 펄스 레이저를 쏘아 반사된 펄스를 센서로 측정하는 방식으로 목표물까지의 거리를 계산한다[30]. 레이저가 목표물에 부딪혔다가 돌아오는 시간과 레이저의 파장을 이용해 목표물에 대한 3차원 디지털 모델을 만든다.

라이다는 1950년대 후반에 처음 개발됐는데 그 후로 서서히 레이저의 활용 범위가 넓어졌다. 라이다를 자율 주행차에서 활용하는 것은 그중 한 예에 불과하다. 해상도가 높고 비금속 객체도 탐지하는 등의 고유한 특성 덕분에 라이다는 3D 지도 제작에 흔히 사용되고 있다. 이러한 속성은 정확한 위치 추정과 내비게이션을 위한 HD 지도에 의존하는 자율 주행차에 적합하다. 그림 2.4는 자율 주행차에 널리 사용되는 벨로다인(Velodyne) HDL-64E 라이다를 보여준다.



그림 2.4. 벨로다인 HDL-64E 라이다(출처: 벨로다인)

이 시스템은 64개의 라이다 채널을 $+2^\circ$ 부터 -24.9° 까지 배치해 26.9° 의 수직 시야를 제공하고 레이저 송/수신부가 회전할 수 있도록 설계해 360° 의 수평 시야각을 제공한다. 회전 속도는 5~20Hz 사이에서 선택할 수 있는데, 이를 통해 라이다 센서가 만드는 관측 지점의 밀도를 사용자가 선택할 수 있다. HDL-64E는 905nm의 파장과 5ns의 펄스의 레이저를 생성해 초당 2200000개의 관측 데이터를 수집할 수 있고, 120m 범위 내의 환경을 $\pm 2\text{cm}$ 의 정확도로 관측할 수 있다. HDL-64E의 상부에는(한그룹에 16개씩 묶은 네 그룹을 배치한) 레이저 발신기가 있고, 하부에는(한그룹에 32개씩 묶은 두그룹을 배치한) 레이저 수신기가 있다.

라이다에서 쏘는 레이저 펄스가 사람 눈에 닿지 않는 것을 보장하기 위해 자동 차용 라이다 빔의 에너지는 눈에 안전한 수준인 클래스 1 레이저 제품으로 엄격히 제한된다[30]. 다른 분야에서는 이보다 훨씬 높은 출력의 빔을 사용한다. 가령



항공 분석용으로는 나무 끝을 충분히 통과해서 숲의 바닥에 닿을 정도로 높은 에너지를 사용한다.

라이다의 기본 작동 과정은 레이더와 비슷하다. 먼저 라이다 장치에서 초당 150,000 펄스의 빠른 속도로 레이저 광선을 목표면에 쏜다. 그런 다음 라이다 장치에 달린 센서로 펄스가 반사되어 돌아올 때까지 걸린 시간을 측정한다. 빛의 속도는 기존에 알려진 값으로 항상 일정하게 유지되기 때문에 목표물과 라이다 장치 사이의 거리를 정확하게 계산할 수 있다. 이 과정을 끊임없이 빠른 속도로 반복하면 라이다가 측정하려는 물체의 표면에 대한 정밀한 ‘지도’를 만들 수 있다.

최신 라이다 제품은 한 번 스캔에 여러 수직 펄스(채널)를 송출해 객체의 높이도 측정하는 기능을 제공한다. 이는 노이즈 필터링이나 객체 인식과 같은 특정한 인지 알고리즘에 유용하게 쓰일 수 있다.

라이다 센서는 세 가지 핵심 요소로 구성된다. 레이저 다이오드(laser diode)는 레이저 빔을 생성하고 포토 다이오드(photo diode)는 반사된 신호를 수신하고, 서보 기반 거울(servo-mounted mirror)은 레이저 빔을 수직과 수평 방향으로 조절한다. 반사된 신호는 포토 다이오드에서 수집해 센서의 신호 처리 유닛에서 처리한다. 센서는 탐지한 객체를 일련의 포인트 클라우드 형태로 출력한다. 각 픽셀마다 측정된 거리와 위치를 센서 기준의 3D 좌표로 표현한다(그림 2.5). 이보다 똑똑한 라이다는 인식한 객체 목록(자동차, 보행자 등)도 제공한다.

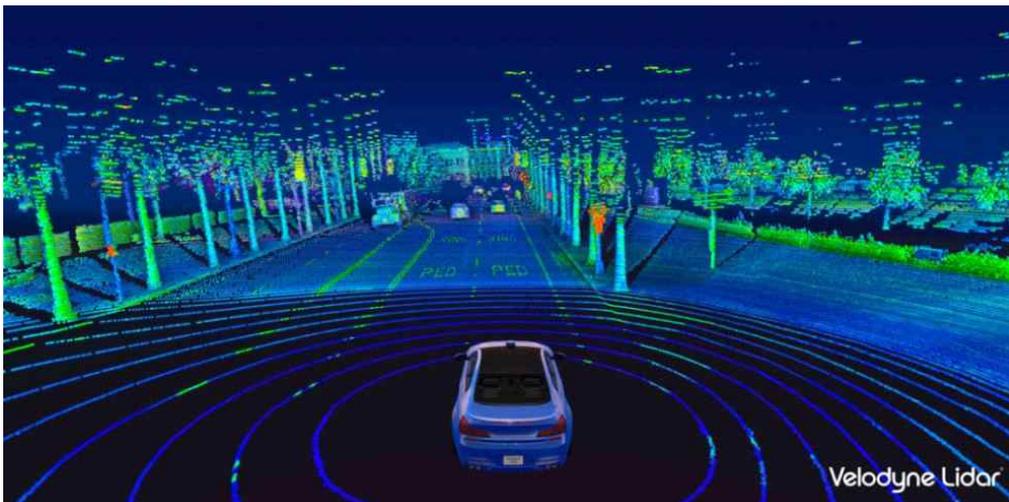


그림 2.5. 라이다 센서가 만들어낸 3차원 포인트 클라우드(출처: 벨로다인)

그림 2.6은 라이다의 핵심 구성 요소와 작동 원리를 보여준다. 생성된 레이저 펄스는 서보 모터로 회전하는 거울을 거쳐 나간다. 이 거울은 펄스를 다양한 수직 각도로 송출할 수 있도록 기울어질 수 있다. 광학 인코더는 거울을 정확히 제어하도록 서보 모터로 피드백을 제공한다. 반사된 신호는(흔히 포토 다이오드 배열로



구성한) 탐지기로 수집해 센서의 신호 처리 유닛을 통해 처리된다.

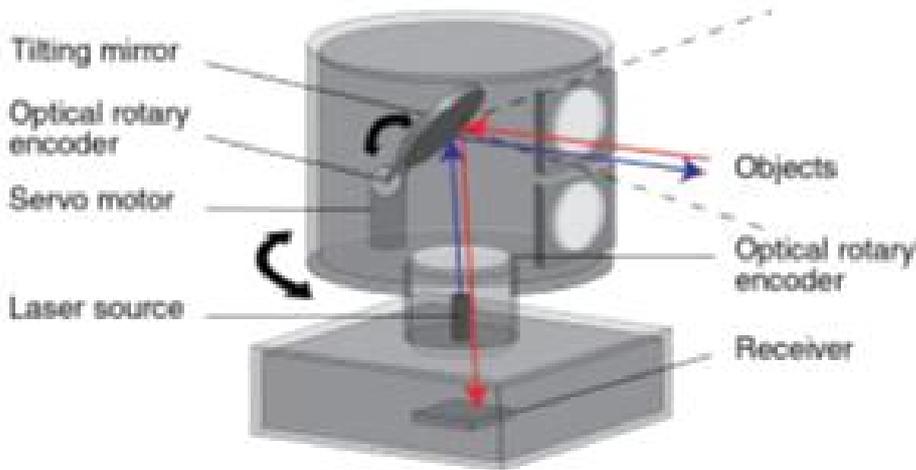


그림 2.6. 라이다의 작동 원리(출처: renishaw.com)

실제로 라이다 제조사와 사용자가 겪는 가장 큰 문제는 캘리브레이션(calibration)이다[31]. 라이다의 성능은 캘리브레이션 성능에 크게 좌우된다. 캘리브레이션을 잘하면 측정된 환경으로부터 정밀한 3차원 데이터를 만들어서 선형(혹은 평면)의 특징을 쉽게 추출할 수 있다. 반대로 캘리브레이션을 제대로 하지 못하면 형상을 추출하기 힘들고 신뢰도가 떨어지며 심지어 형상 자체를 추출하지 못할 수도 있다.

라이다는 빔폭(beamwave)이 좁고 다소 먼 거리까지 도달해서 고해상도 3D 맵핑에 적합한 센서로 자리 잡았다. 또한 라이다는 실내 측위(indoor positioning)와 같이, 위성 기반 GNSS를 사용할 수 없는 영역에서 또 중요한 역할을 한다. 라이다는 수신한 적외선 빛의 세기를 측정하기 때문에 낮과 빔을 안정적으로 감지하는데 활용할 수도 있다. 태양이 발산하는 적외선의 양은 낮에 생성한 레이저 빔보다 훨씬 많기 때문이다.

라이다는 자동차 분야에서 유용하게 사용되고 있지만, 여전히 가격이 높아서 양산 차에 널리 보급하기에는 큰 장벽이 되고 있다. 하지만 현재 개발 중인 고정형 라이다가 나오면, 센서의 비용과 크기가 크게 줄어들 것으로 기대하고 있다.

레이저 빔은 안개나 먼지 등과 같은 작은 입자에도 반사하기 때문에 환경에 상당히 민감하며, 날씨가 안 좋은 날에는 레이더에 비해 노이즈가 많이 생성된다. 그래서 라이다 센서를 자동차에 적용하는 것은 레이더의 경우보다 더 복잡하다. 필터링 알고리즘으로 눈발이나 빗방울에 의한 간섭을 어느 정도 줄일 수 있지만, 센서 표면에 붙어있는 먼지, 얼음, 눈에는 효과가 크지 않다. 라이다를 앞 유리 안쪽에 두면 이런 문제를 어느 정도 해결할 수 있지만, 와이퍼 움직임에 영향을 받고, 차량 내부에서는 360도 범위로 인지 작업을 수행할 수 없으며 비전 카메라



나 레인 센서 같은 다른 센서와의 충돌이 발생할 가능성이 있다.

2.2.2. HD 지도 개요

이번에는 HD 지도를 만드는 기술을 자세히 살펴보자. 그 전에 먼저 자율 주행에 HD 지도가 필요한 이유부터 이해할 필요가 있다. 집에서 사무실로 가는 익숙한 길을 주행하는 경우를 생각해보자. 주행하기 전에 이미 머릿속에 경로를 담고 있기 때문에 주행 과정에서 특별히 안전에 유의해야 할 지점에 집중할 수 있다. 예를 들어 최근 몇 주 사이에 생긴 도로의 패인 곳과 같은 장애물을 충분히 예상할 수 있고, 큰 트럭이 속도 제한 표지판을 가리더라도 제한 속도를 알고 있을 것이다. 이와 달리 처음 가보는 길을 운전할 때처럼 경로에 대한 정보가 훨씬 부족한 상황에서는 좀 더 많은 주의를 기울여야 한다. 그래서 주행하는 순간마다 적절히 반응할 수밖에 없다. 자율 주행차도 마찬가지다. 자율 주행차가 좀 더 익숙하게 주행해서 안전에 집중할 수 있도록 HD 지도를 제공한다. 그렇다면 기존에 나와 있는 상용 디지털 지도를 활용하면 되지 않느냐고 생각할 수 있다. 이러한 상용 디지털 지도는 사람에 맞추어 만들어졌기 때문에 해상도가 미터 급으로 낮은 편이고, 정보의 업데이트 주기도 길다. 자율 주행차가 도로를 주행하려면 도로에 대한 구체적인 정보가 필요하다. 자동차가 특정한 차로를 따라 주행하려면 차로의 위치를 정확히 알아야 한다. 따라서 위치 추정을 데시미터 단위로 정확히 수행해야 한다. 결론적으로 상용 디지털 지도만으로는 자율 주행차가 실시간으로 위치 추정을 수행하기가 굉장히 힘들다. 특히 지도에 나온 정보와 실제 환경이 일치하지 않으면 더 힘들다.

자율 주행차를 위한 지도를 제작하기 위해 해결해야 할 중요한 문제는 크게 세 가지가 있다. 첫째 지도의 정밀도가 센티미터급이어야 한다. 따라서 HD 수준으로 만들어야 한다. 둘째 도로의 변화를 반영하도록 지도를 업데이트해야 한다. 실제로는 일주일에 한 번 업데이트한다. 셋째 자율 주행 시스템이 높은 성능을 유지하도록 다른 모듈과 매끄럽게 연동되어야 한다. 라이다와 다른 센서를 결합하면 정밀도를 높일 수 있다. 최신 정보를 반영하도록 딥맵(DeepMap)에서 사용하는 클라우드 소싱 방법을 적용할 수도 있고 이와 반대로 구글이나 바이두에서 사용하는 측정 차량을 통한 주기적인 업데이트 방법을 적용할 수도 있다. 이렇게 만든 HD 지도를 자율 주행 시스템의 다른 모듈과 매끄럽게 연동하려면 HD 지도를 실시간으로 업데이트해서 자율 주행 자동차로 제공하는 고성능 자율 주행 클라우드 인프라스트럭처를 구축해야 한다. 센티미터급의 정밀도를 갖는 HD 지도를 제작하는 방법에 대해서는 다음 절에서 소개한다.

HD 지도를 제작하는 핵심 기술은 주변 환경의 세밀한 지도를 만들어서



GNSS/INS 항법을 보완한 뒤 자동차에 탑재된 라이다 센서로 HD 지도 기준의 자동차 위치를 알아내는 것이다[32, 33, 34]. 따라서 HD 지도 제작의 핵심은 여러 센서 (GNSS/INS/LiDAR)를 융합해서 지도의 한 격자에서 발생하는 오차를 최소화하는 데 있다. 이때 먼저 GNSS/INS로 매번 스캔할 때마다 개략적인 위치 정보를 생성한 다음, 라이다로 2차원 위치를 좀 더 정밀하게 측정한다. 여기서 가장 중요한 부분은 여러 센서를 조합해 고정밀 로컬 지도를 만들고 이렇게 만든 로컬 지도를 연결해 글로벌 지도를 만드는 것이다.

2.2.3. HD 지도의 구조

먼저 HD 지도의 구조를 자세히 살펴보자. 기존 지도와 마찬가지로 HD 지도도 계층적인 데이터 구조를 갖는다. HD 지도는 통상 다수 계층을 갖고 자율 주행차를 위한 충분히 누적된 정보를 제공한다. 모든 계층의 사이즈 때문에 보통은 클라우드로부터 자율 주행차에 서비스되고[35, 36], 부근의 매우 작은 구역의 HD 지도 (이를 서브 지도라 함)만이 필요할 때 자동차에 다운로드 된다.

HD 지도의 계층들은 서로 많이 다르고, 서로 다른 표현, 데이터 구조 및 목적을 갖고 있다. 비록 HD 지도 구축자(builder)가 반드시 동일한 관례를 따라야 하는 것은 아니지만, 통상적으로 HD 지도는 다음과 같은 4개 계층을 포함한다.

2.2.3.1. 2차원 정사영 반사 지도(Orthographic Reflectivity Map : ORM)

ORM은 도로표면의 서로 다른 물질들(즉 도로포장, 도로마킹 페인트 등)이 레이저로부터의 상이한 적외선 반사 세기를 갖는다는 사실을 활용한 것이다. 이 계층은 라이다 센서의 포인트 클라우드로부터 추출된 도로 표면의 2차원 평판시야이다. 실제로 반사 지도는 동일한 구역의 복수스캔을 결합하고 반사신호 세기값으로 텍스처 매핑(물체의 표면에 세부적인 질감의 묘사를 하거나 색을 칠하는 기법)을 처리하고 나면 사실적인 것처럼 보일 것이다.

2.2.3.2. 디지털 고도 모델(Digital Elevation Model: DEM)

DEM은 3차원 모델이고, 도로연석의 높이, 램프 혹은 기복 있는 도로의 경사도/급준함 등과 같은 주행환경의 표면에 관한 높이 정보를 포함한다. 이는 위치추정(특정이 부족한 도로표면을 갖는 상황), 움직임 계획과 자동차 제어에 유용하다.

2.2.3.3. 차선/도로 모델

차선/도로 모델은 차선 및 도로 세그먼트의 시맨틱스를 포함하는 매우 중요한 벡터화된 계층이다. 도로모델은 도로의 에지와 같은 차선부분이 아닌 도로부분을



포함한다. 그러나 자율 주행차 설계자들은 항상 자율주행차가 차선중앙이 되도록 시도하지만, 실제로는 차선경계 밖으로 운행하는 것이 필요한 희소한 경우를 제외하고 차선모델만을 다루는 것이 필요하다. 차선모델은 차선기하(경계, 넓이, 곡률 등), 차선타입(자동차선, 바이크선, 버스전용선 등), 차선방향, 차선마킹/분리 타입들(실선 대비 점선, 단일 대비 두 개(double) 등), 제한(좌/우회전만), 속도제한, 차선간의 연결성 등에 관한 정보를 포함한다. 차선/도로 모델은 움직임 계획과 자동차 제어 등에 핵심적이다.

2.2.3.4. 정적 지도(Stationary Map)

이는 잘 정의되지 않은 계층이다. 다른 계층에서 포착되지 않은 운전환경 내 정적요소들(즉 교통신호등과 차선, 도로장애물 등과의 연관성)을 저장하는 다방면의 계층이다.

2.2.4. HD 지도 생성

그림 2.7에서 보여주고 있는 것처럼, HD 지도 생성과정은 다음과 같은 4가지 단계로 나눌 수 있다 : 데이터 수집, HD 지도 생성, 품질제어와 유용성, 갱신 및 유지

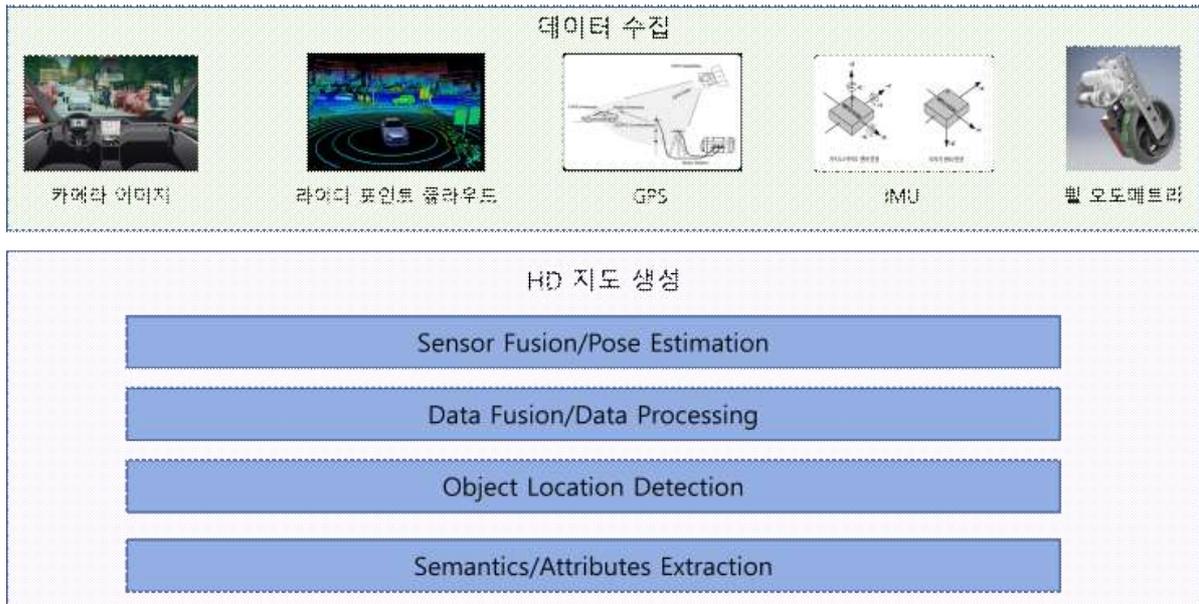


그림 2.7. HD 지도 생성

2.2.4.1. 데이터 수집

모바일 맵핑 시스템(MMSs)은 라이다, 카메라, GPS, IMU 및 휠 주행거리계 등과 같은 다중 센서들이 장착된다. 그리고 데이터를 모으기 위해 현장으로 가서 이 데이터들을 저장장치에 기록하거나, 필터링이나 압축과 같은 어느 정도의 데이터 처



리 후 통신망을 통해 서버나 클라우드 저장장치로 데이터를 보낸다. 데이터수집 처리는 보통 도시 영역으로 수행되고 최적 데이터 저장 및 전송 뿐만 아니라 세심한 라우터 계획을 포함한다. 장비 가격, 노동력, 데이터 저장 및 전송은 데이터 수집 처리의 주요 관심사들이고, 같은 도로 부분의 재수집과 같은 오버드라이버 감소는 실행자의 큰 관심사이다.

HD 지도 생성용으로 수집된 데이터는 두 가지 범주에 속한다. (i) HD 지도 데이터 : 라이다 포인트 클라우드와 HD 지도의 내용이 될 기하 및 시맨틱 정보를 포함하는 카메라 이미지, (ii) 보조 데이터 : HD 지도를 위한 기하 및 시맨틱 정보는 포함하지 않지만 HD 지도 생성에 유용한 GPS/IMU/휠 주행거리계의 이력, 보조 데이터의 사용은 주로 데이터 수집 차량용 방위 최적화를 위한 것이다.

2.2.4.2. HD 지도의 오프라인 생성

이는 수집된 데이터를 처리하고 HD 지도를 생성하는 후방지원 업무용(back-office) 작업이다. 개략적으로 4단계로 세분화할 수 있다(그림 2.7).

2.2.4.2.1. 센서 융합과 방위 추정

MMS 차량의 정확한 방위(위치와 방향)을 아는 것은 HD 지도 생성에 핵심이다. 차량의 방위가 정확하지 않으면, 정밀한 지도를 생성하는 것은 불가능하다. 일단 데이터 수집 차량의 정확한 방위를 갖고, 장착된 센서의 위치 및 차량 프레임에 대한 상대적인 각도 등이 주어지면 수집된 포인트 클라우드와 이미지 프레임의 정확한 방위를 추론할 수 있을 것이다.

비록 정확한 방위는 GPS, IMU 및 휠 주행거리계의 제한으로 인해 수행 시간에서 직접적으로 얻을 수 없지만, 그래프 기반 SLAM의 사용으로 상이한 센서들의 이력 융합과 같은 오프라인 최적화를 이용함으로써 정확한 방위를 추정할 수 있다[33, 37].

2.2.4.2.2. 지도 데이터 융합과 데이터 처리

일단 정확한 방위를 갖고 난 다음에 지도 데이터 융합을 수행할 수 있을 것이다. 여기서 지도 데이터는 라이다 3차원 포인트 클라우드와 카메라 이미지를 의미한다. HD 매핑을 위해, 비디오의 해상도와 품질이 보통 만족스럽지 않고, 비디오의 높은 프레임 레이트를 필요로 하지 않는다. 10FPS 이하에서 획득한 보다 높은 해상도의 이미지가 공통적으로 널리 사용된다. 데이터의 융합처리 동안 밀집 포인트 클라우드를 얻기 위해 다중 스캔의 포인트 클라우드들이 정렬하고 캘리브레이트되어야 한다. 또한 포인트 클라우드와 카메라 이미지는 각각 등록되어, 사물의 3차원 위치를 얻는데 포인트 클라우드를 사용하거나 시맨틱 정보 추출에 등록된



이미지를 사용할 수 있어야 할 것이다. 이는 비록 포인트 클라우드의 정확한 3차원 위치를 제공하지만 시맨틱 정보를 제공할 수 없고 반면에 이미지는 정확한 시맨틱 정보를 제공하기 때문이다.

더불어서 도로평면 생성 무관한 사물제거(즉 동적사물이나 도로에서 멀리 떨어진 사물 들), 사실적인 정사영 이미지 (photorealistic orthographic image) 생성을 위한 텍스처링 등과 같은 다른 데이터 처리 단계들도 수행된다.

2.2.4.2.3. 3차원 사물위치 검출

기하 및 정밀한 위치가 중요한 도로 요소들(차선경계, 보행자도로의 연석, 교통 신호등, 추월, 철도트랙, 가드레일, 가로등 기둥, 속도제한 범퍼, 포트홀 등)을 위해, 이들의 정밀한 3차원 위치를 갖는 지도가 필요하다. 라이더 포인트 클라우드는 직접적으로 3차원 위치 정보를 제공하고, 포인트 클라우드에 의한 3차원 사물의 검출은 기하 기반 방법 [38-41] 또는 3차원 포인트 클라우드의 심화학습[42-44]의 사용으로 수행된다. 물론 같은 사물의 복수개 이미지를 사용한 삼각기법을 통해 포인트 클라우드를 사용하지 않고 3D 사물의 위치를 검출할 수 있다.

2.2.4.2.4. 시맨틱/속성 추출

마지막이지만 가장 힘든 단계가 HD 지도를 위해 데이터로부터 시맨틱과 속성을 추출하는 것이다. 이 처리에는 차선/도로 모델 구성, 교통신호 인식, 교통신호등의 차선과의 결합, 도로마킹 시맨틱스 추출 도로 요소(즉 가로등, 가로등 기둥) 등이 포함된다.

2.2.5. 동적 지도(LDM, Local Dynamic Map) 생성

유럽 표준화 기구(ETSI)에서 정의한 동적 지도(Local Dynamic Map)는 시간의 흐름에 따라 변화하는 도로환경을 지도에 표현하고 갱신하기 위해 제안되었으며 동적 지도는 차량 및 인프라에서 수집되는 동적/정적 정보를 저장하여 차량에 제공하는 기술이다. 자율주행차의 주행 경로 및 주변 환경에 대한 지형 정보, 위치 정보, 상태 정보 등을 전달하고, 자동차의 차체 센서나 교통정보를 제공받아 정보를 수집하고 저장한다. 동적 지도의 계층 구조에 대해서는 아직 논의 및 개발 중이지만 일반적으로 LDM은 그림 2.8과 같이 4가지 정보 계층으로 구성된다.



그림 2.8. LDM 개념도(출처: 현대 오토에버)

1계층은 영구적인 정적 데이터를 포함하며, 주로 지형을 포함한 지도를 의미한다. 2계층은 운전 중에 얻을 수 있는 일시적인 정적 데이터를 포함하며, 예를 들어 휴게소 운영 시간, 도로시설물 등의 정보를 저장한다. 3계층은 일시적인 동적 데이터를 포함하며, 사고, 정체, 날씨와 같은 교통 정보와 기상 상황 정보를 제공한다. 4계층은 가장 동적인 데이터를 포함하며, 차량의 위치, 속성, 상태 등의 정보를 저장한다.

2.2.6. 라이다와 HD 지도를 이용한 위치 추정

HD 지도를 만들었다면 HD 지도를 활용해 자동차의 위치 추정을 수행해야 한다 [45-48]. 일반적으로 위치 추정 과정에 파티클 필터를 적용하는데, 우선 파티클 필터로 거리(range) 데이터를 분석해 자동차 아래의 지평면을 추출한다. 그런 다음, 이와 측정된 적외선 반사 지도 정보에 대한 상관 함수를 피어슨 적률 상관 함수(Pearson product-moment correlation)를 통해 계산한다. 각 파티클은 시간에 대한 속도를 투영한 값을 갖는다. 속도는 휠 주행거리계, INS, GNSS를 사용하는 강 결합(tightly coupled) 관성 항법 시스템을 통해 추정한다.



HD 지도에 대한 위치 추정에 깊이 들어가기 전에 먼저 위치 추정에 필요한 수학적 도구부터 이해하고 넘어가자. 파티클 필터를 널리 사용하기 전에는 주로 칼만 필터로 상태 공간 모델(state space model)을 풀었다. 칼만 필터(Kalman filter)는 선형 가우시안 상태 공간 모델(Linear Gaussian state space model)에 대한 최적해를 구하는 필터이다. 선형성 혹은 가우시안 조건을 만족하지 않는 환경이라면 확장 칼만 필터(Extended Kalman Filter, EKF)나 무향 칼만 필터(Unscented Kalman Filter, UKF)를 적용한다. 하지만 비선형성이 크고 가우시안 조건을 만족하지 않을 때는 칼만 필터로 추정한 결과를 신뢰할 수 없다. 이러한 환경에는 파티클 필터 기술을 적용하는 것이 좋다. 파티클 필터는 관측할 수 있는 시점부터 실시간으로 동작해 잠재 프로세스에 대한 주변 분포(marginal distribution)의 근사치를 구한다. 파티클 필터 알고리즘을 수행할 때 중요도 샘플링(Importance sampling) 과정을 통해(‘파티클’이라고 부르는) 이산 값과 그 값에 대한 비중의 집합으로 각 시점마다 분포의 근사치를 구한다[49]. 칼만 필터는 선형성과 가우시안 분포에 대한 가정에 의존한다. 특정한 파티클 필터에서는 칼만 필터의 비선형 및 (혹은) 비가우시안 환경에 대한 작업을 순차 몬테 카를로 방법(Sequential Monte Carlo methods)으로 재현한다. 두 필터의 핵심적인 차이는 칼만 필터가 시스템 모델을 유도해 사용하는 반면, 파티클 필터는 상태와 이노베이션의 추정치를 시뮬레이션을 통해 만드는 데 있다. 선형 가우시안 모델에 대해 파티클 필터를 적용하면 칼만 필터와 동일한 가능도(likelihood)를 얻을 수 있다. 연산의 관점에서 칼만 필터는 시뮬레이션을 하지 않기 때문에 선형 가우시안 모델보다 계산량이 상대적으로 적다. 정리하면, 선형 모델을 통해 시스템을 적절히 묘사할 수 없거나 센서의 불확실성이 가우시안 분포를 따르지 않는다고 하더라도 파티클 필터는 모든 모델을 충분히 많은 파티클로 이산화하는 것을 통해 확률 분포를 추정한다(이때 각 파티클이 상태 공간 모델의 역할을 한다).

파티클 필터는 HD 지도의 위치 추정 과정에서 실시간으로 적용된다. 그림 2.9는 파티클 필터의 위치 추정 과정을 보여준다[50]. 왼쪽 그림을 보면 초기 파티클이 뿌려져 있다. 오른쪽 그림은 생성된 지도와 차량의 이동 경로를 보여준다. 노란색 마커는 파티클, 청색 마커는 라이다로 추출된 차선, 적색 마커는 GPS 위치, 초록색 마커는 파티클 필터로 추정도니 차량의 위치를 나타낸다. 파티클 필터 알고리즘은 예측과 업데이트라는 두 단계를 반복한다. 자동차가 움직일 때마다 파티클을 모델에 따라 이동시키는데 이때 랜덤 오차를 반영해 추정 변수의 오차에 대한 효과를 나타낸다. 이 과정을 예측 단계(prediction stage)라 부른다. 그런 다음 각 파티클의 비중을 최신 라이다 스캔 데이터로 재평가를 하는 과정을 업데이트 단계(update stage)라 부른다. 특히 라이다 스캔 데이터가 파티클이 분포한 영역 주변에 형성되면, 파티클이 자동차의 실제 위치와 근접한다는 뜻이므로 비중이 매우



높아진다. 업데이트 과정 이후 좁은 영역에 높은 비중을 갖는 파티클이 몰려 있다면(수렴 상태), 위치 추정의 정확도를 더 올리기 위해 좁은 영역에 대해 전체 알고리즘을 다시 수행하는데, 이를 리샘플링(resampling)이라고 부른다.

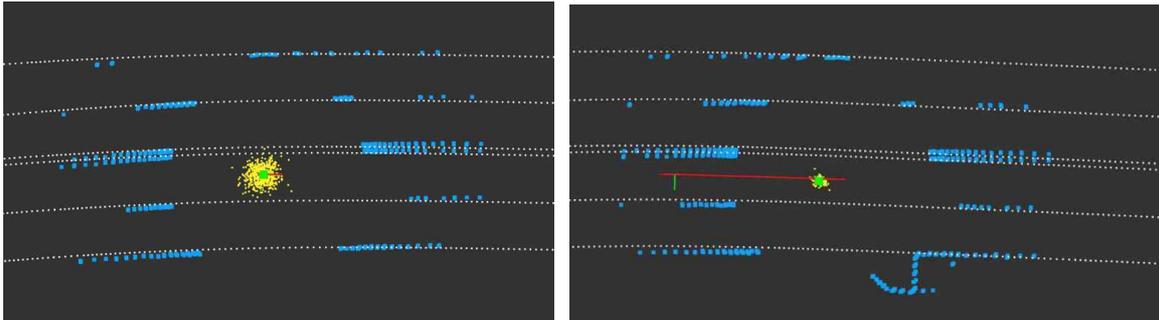


그림 2.9. 파티클 필터의 동작(출처: [50])

파티클 필터는 실시간 성능을 내기 위해 3차원 자세 벡터(x , y , yaw)만을 다루고, 롤(roll)과 피치(pitch)는 충분히 정확하다고 가정한다. 파티클 필터를 통해 추정하는 움직임은 관성 속도 측정치에 기반한다. 라이다 측정 데이터에서 지표면과 관계없는 요소들은 매핑 과정 중 지평면 분석 과정에서 제거된다. 또한 라이다 측정 데이터는 현재 환경에 대한 지도가 만들어진 다음에 저장되는데 따라서 사전 반사 값(prior reflectivity value)이 지도에 포함된다. 위치 추정 오차를 줄이려면 현재 GNSS/INS 자세 추정치로부터 적은 숫자의 파티클을 지속적으로 만들어야 한다. GNSS를 사용할 수 있다면, 측정 가능도(measurement likelihood)의 계산에 GNSS 정보를 적용해 GNSS의 위치 추정치와 파티클의 거리가 너무 멀어지지 않게 해야 한다. 자동차의 위치 추정을 복잡하게 만드는 요인 중 하나는 바로 날씨다. 비로 인해 젖은 표면은 마른 표면보다 적외선 빛을 덜 반사해서 도로의 가시성이 줄어든다. 파티클 필터는 거리를 측정할 때마다 밝기와 표준편차를 정규화해 이러한 문제에 대응한다. 이 과정은 측정 환경의 로컬 지도를 만들 때도 적용된다. 피어슨 적률 함수와 참 값을 모르는(missing) 변수의 상관 값을 계산할 때도 정규화된 최소자승법(least squares difference method)을 사용한다.

지금까지 라이다를 통한 HD 지도의 위치 추정 기초에 대해 설명했다. 이번에는 위치 추정 과정에서 발생하는 문제점을 살펴보자. 첫 번째 문제는 위치 추정 오차다. 앞에서 라이다 스캔과 HD 지도의 일치 여부를 이진 분류 방식으로 판단했다. 만약 스캔 데이터와 지도가 일치하지 않으면 해당 라이다 스캔 데이터를 폐기한다. 이 방법은 간단하고 큰 문제가 발생하지는 않지만 실제로 큰 오차가 발생할 수 있다. 이진 분류 방법을 개선하기 위해 [45]의 저자는 HD 지도를 확장하고 환경의 확률적 특성을 캡슐화하는 방식을 제안했다. 이를 통해 세계를 좀 더 정확하게 표현하고, 위치 추정 성능을 크게 높일 수 있다. 저자는 환경과 라이다 스캔



데이터의 일치 여부를 직접 확인하지 않고 관측 데이터를 모두 더하는 방법을 사용했고, 로컬 지도의 측정 분산을 모델링했다. 이 방법은 앞서 언급한 비확률론적 방법과 비교했을 때 몇 가지 장점이 있다. 측정할 지표면이 (입사 경로와 반사 경로가 같은) 역반사 특성을 가질 때 표면의 반사 정도(remission)가 입사각에 상대적으로 영향을 적게 받는 뜻밖의 현상이 발생해서 위치 추정의 오차가 줄어든다. 하지만 측정하는 지표면이 빛나는 물체 등으로 인해 각반사 특성이 나타나면 측정 위치가 다를 때 결과가 일정하지 않다. 이러한 차이를 무시하면 위치 추정 오차가 커지지만, [45]에서 제안한 방법은 이러한 점을 모두 고려해서 위치 추정 오차를 줄인다. 또한 이 방법은 움직이는 장애물에 대해 높은 견고성(robustness)을 제공한다. HD 지도의 반사 측정치 분포를 모델링함으로써 지도에 나온 경로에 있는 움직이는 장애물로 인한 영향을 자동적으로 제거할 수 있다. 이 방법의 세 번째 장점은 위치 추정에서 사용되는 측정치 모델에 대한 확률적 해석을 직접 할 수 있다는 점이다.

라이다 기반의 위치 추정에서 발생하는 두 번째 문제는 라이다 센서가 비싸다는 것이다. 3차원 라이다 장비는 한 대의 가격이 대략 100000달러에 달한다. 이렇게 높은 가격은 자율 주행 자동차의 상용화를 방해하는 주된 원인 중 하나다. 이 문제를 해결하기 위해 가격 대 성능비가 좋은 센서를 최적화하는 다양한 방법이 제안되고 있다. 특히 [46]에서 저자는 고가의 라이다 장비를 통해 미리 만들어둔 HD 지도를 활용하되, 푸시브룸(push-broom) 2차원 스캐너 한 대를 사용해 도로상의 자동차에 대한 정밀한 위치 추정을 수행하는 방식을 제안했다. 여기서 2차원 레이저 스캐너는 아래를 향해 발사하도록 조절되어 레이저가 연속적으로 지표면에 닿는다. 이러한 방식을 통해 라이다 데이터의 작은 조각을 만들 수 있고 통계적으로 HD 지도와 매칭할 수 있다. 이 기술을 적용하기 위해서는 자동차의 속도계에서 얻을 수 있는 정확한 속도 정보를 실시간으로 위치 추정 모듈에 제공할 필요가 있다. 저자는 이 방식을 통해 정확한 GNSS/INS 위치 추정 시스템을 구축하는 데 성공했다. [47]에서도 비슷한 대안을 제시했다. 이 논문의 저자는 2차원 라이다와 주행거리계 정보만으로 자동차의 3차원 도시 환경에 대한 위치 추정을 수행하는 알고리즘을 고안했다.

눈, 비 같은 날씨 또한 라이다 기반의 위치 추정에서 문제 요인이 되는데, 라이다의 성능이 기상 조건에 크게 영향을 받기 때문이다. 예를 들어, 악천후로 인해 도로의 페인트에 대한 시야가 차단되거나 도로 표면의 품질이 나빠지면 라이다 기반으로 위치 추정을 제대로 할 수 없다. 악천후 환경은 라이다가 스캔한 반사 값에 악영향을 미치는데 만약 위치 추정에 반사 대신 다른 정보를 활용한다면 이 문제를 해결할 수 있다. [48]에서 저자는 환경과 측정 데이터의 매칭에서 반사 대신 장면의 3차원 구조를 활용하는 방식을 제안했다. 이 방식을 적용하기 위해 저

자는 환경의 구조를 가우시안 혼합(Gaussian mixture) 지도 형태로 전환했다. 이때 지도는 z 높이 분포에 대한 가우시안 혼합의 집합체와 같은 형태를 갖는다. 실시간성을 확보하기 위해 저자는 가우시안 혼합에 대한 래스터화 해석 테이블(rasterized lookup table)을 사용하는 분기 한정 다해상도(branch-and-bound multi-resolution) 접근법을 개발했다.

2.3. 시각적 주행거리계

시각적 주행거리계(visual odometry, VO)는 하나 이상의 카메라에서 입력된 데이터만으로 자동차의 움직임을 추정하는 과정이다[51-53]. 비주얼 오도메트리에서 수행하는 주된 작업은 두 이미지 사이의 상대적인 변화를 구한 다음, 이를 통해 자동차의 궤적(trajjectory)을 구하는 것이다. 즉, 시각적 주행거리계로 차량의 자세가 변할 때마다 경로를 점진적으로 복원한다. 그런 다음 좀 더 정확한 로컬 궤적의 추정치를 구하기 위해 최근 자세에 대한 교정 작업을 반복한다. 이러한 반복 교정(iterative refinement)은 최근 m 개의 이미지에 대해 재구성한 3D 지점들의 재투영 오차의 제곱합을 최소화하는 방식으로 진행되는데 이를 번들 조정(Bundle Adjustment)이라 부른다.

일반적인 시각적 주행거리계의 진행 과정은 그림 2.10과 같다.

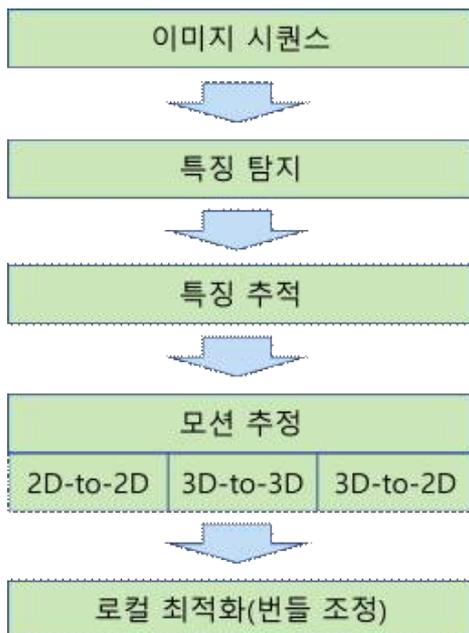


그림 2.10. 비주얼 오도메트리의 파이프라인[30]

새로 들어온 이미지마다 다음과 같은 세 단계를 거친다. 첫 단계는 특징점(feature point)을 추출한다. 두 번째 단계는 이렇게 추출된 특징점과 일치하는 부분이 이전 프레임에 있는지 검사한다. 이때 각기 다른 프레임에서 공통되는 3차원



특징을 2차원 특징으로 재투영하는 것을 이미지 대응(image correspondence)이라 부른다. 세 번째 단계는 모션 추정(motion estimation)을 수행한다. 이 작업은 두 이미지가 입력된 시간 $x-1$ 과 x 사이의 상대적인 움직임 T_x 를 계산한 뒤, 결과로 나온 T_x 와 이전 자세를 결합해 자동차의 현재 자세 V_x 를 구한다. 마지막으로, 최신 x 프레임에 대해 번들 조정을 수행해 로컬 궤적 추정치의 정확도를 높인다.

모션 추정은 VO 시스템에서 각 이미지마다 수행하는 핵심 연산 단계다. 좀 더 정밀하게 설명하면, 모션 추정 단계에서 현재 이미지와 이전 이미지 사이에 발생한 자동차의 움직임(모션)을 계산한다. 그리고 이러한 모션을 모두 합쳐서 자동차의 전체 궤적을 복원한다. 특징을 2차원에서 대응했는지 아니면 3차원에서 대응했는지 여부에 따라 세 가지 방식으로 나뉜다.

- 3D-3D 대응: 이 경우, 카메라 모션(변환)은 3차원으로 지정된 두 세트의 대응물로부터 계산될 수 있다. 먼저, 두 개의 스테레오 이미지 쌍을 캡처하여 그들 사이의 특징점을 추출하고 일치시킨다. 둘째, 각 스테레오 쌍에 대해 3차원 일치 지점을 삼각 측량한다. 변환은 두 3차원 포인트 세트 사이의 L2 거리를 최소화하여 절대 척도로 계산된다[30].

- 2D-2D 대응: 이 방법에서 변환 행렬은 필수 행렬[51]을 사용하여 계산된다. 필수 행렬은 두 순차적 이미지 사이의 기하학적 관계를 부정하고 에피폴라 제약 조건을 사용하여 2차원 특징 대응에서 계산된다. 필수 행렬을 계산하는 간단하고 일반적인 접근법은 Nister five-point 알고리즘을 사용하는 것이다[54]. 이 방법에서는 연속된 프레임 간의 상대적 척도를 결정하기 위해 해당 점 5개의 세트가 사용됩니다. 필수 행렬을 계산하는 또 다른 방법은 Fischler와 Firschein이 제시한 8점 알고리즘을 사용하는 것이다[55]. 이 접근법의 주요 문제는 알려지지 않은 스케일 팩터까지 변환 행렬을 계산한다는 것이다.

- 3D-2D 대응: 이 방법의 주요 개념은 2D 및 3D 대응에서 2D 재투사 오류를 최소화하여 변환 행렬을 계산하는 것이다. 카메라 자세를 복구하기 위한 최소 솔루션은 원근법-3점(P3P)으로 알려진 세 개의 3D-2D 대응이 필요하다[56]. 3D-2D에 기반한 움직임 추정은 3D-3D 특징 위치 오류[54] 대신 영상의 재투사 오류를 최소화하여 3D-3D보다 더 나은 정확도를 제공한다.

시각적 주행 기록계는 알고리즘에 사용되는 카메라의 유형과 개수에 따라 분류할 수 있다.

2.3.1. 스테레오 시각적 주행 기록계

관성 센서(inertial sensor)는 한쪽으로 흐르는 드리프트(drift) 현상이 발생하기



쉽고, 휠 주행 기록계(wheel odometry)는 울퉁불퉁한 오프로드(off-road; 비포장) 지형에서는 측정 값이 불안정하기 때문에, 실시간 위치 추정에 비주얼 오도메트리를 활용한다. 스테레오(stereo; 이중) 비주얼 오도메트리는 연속적으로 들어오는 스테레오 이미지 쌍으로부터 프레임 단위로 카메라의 움직임을 추정하는 방식으로 작동하는 기술로 자율 주행에 널리 적용되고 있다. 예를 들어 [57]에서 제시한 알고리즘은 흔히 사용되는 비주얼 오도메트리 알고리즘과 다음과 같이 크게 두 부분에서 차이가 있다. 첫째, 카메라 움직임에 대해 사전에 어떠한 가정도 하지 않는다. 둘째 별도의 스테레오 알고리즘으로 계산한 디스패리티(시차, 양안차) 이미지(disparity image)를 기반으로 추정한다. 이 논문의 저자는 400m를 주행하는 동안 수집한 4,000프레임의 이미지에 대해 위치 오차가 (주행 거리의 0.25%에 해당하는) 1m 미만임을 보여줬다. 그리고 처리 시간은 512 x 384 이미지에 대해 20ms에 불과했다. 이 알고리즘의 기본 작동 과정은 다음과 같다. 첫째, 코너 특징 탐지(corner feature detection) 방식으로 각 프레임마다 특징을 검출한다. 둘째, 로컬 윈도우에 대한 절대 합을 통해 프레임 사이의 특징을 매칭한다. 셋째, 일관된 매칭 집합 중 가장 큰 것을 찾는다(인라이어 검출; inlier detection), 넷째, 인라이어 집합에 있는 특징에 대한 재투영 오차를 최소화하는 프레임 사이의 움직임(모션)을 찾는다. 특징 매칭 단계에서 부정확하게 대응되는 결과가 나오는 것을 피할 수 없는데, 그대로 내버려두면 프레임 사이의 모션 추정 값이 바람직하지 않은 방향으로 치우치게 된다. 이 문제를 해결하는 일반적인 방법은 매칭 오류가 발생하더라도 큰 영향을 받지 않도록 측정 알고리즘을 견고하게 만드는 것이다. 하지만 [57]에서 설명한 바에 따르면 인라이어 검출 단계에서 스테레오 데이터를 활용하는 방식을 사용했다. 이 방법은 특징점의 3D 위치는 강제성(rigidity) 조건을 반드시 지켜야 하며, 이러한 제약사항은 프레임 사이의 모션 추정치를 계산하기 전에 상호 일관된 특징 집합을 식별하는 데 활용할 수 있다는 점에 착안했다. 이 알고리즘은 초기 모션 추정치가 없어도 되므로, 굉장히 큰 이미지 전환 작업도 처리할 수 있다는 점이 주목할 만하다.

2.3.2. 모노컬러

모노컬러 시각적 주행 거리계는 스테레오 시각적 주행 거리계와 달리, 상대적인 움직임과 3차원 구조를 모두 2차원 데이터로 계산해야 한다. 이는 절대적인 깊이에 대한 차이를 알 수 없기 때문이다. 첫 두 카메라 자세 사이의 거리를 주로 고정된 값으로 설정해서 초기 3차원 구조를 만든다. 새 이미지가 들어오면 3D 구조에 대한 정보를 이용해 처음 두 프레임의 상대적인 스케일과 카메라 자세를 결정한다. 스테레오 카메라 설정에서와 달리 모노컬러 시각적 주행 거리계는 기준 문



제로 어려움을 겪지 않으며, 그것이 최근 몇 년 동안 모노클러 타입이 많은 관심을 끌고 있는 이유이다. 그러나, 3차원 정보를 재구성하기 위해서는 적어도 3개의 다른 프레임이 필요하다[58]. 더욱이, 모노클러 타입 접근법의 주요 단점 중 하나는 처음 두 프레임 사이의 변환(방향 및 변환)이 완전히 알려져 있지 않기 때문에 변환 벡터가 상대적인 규모까지 계산된다는 것이다. 이 문제를 해결하는 한 가지 방법은 IMU와 라이다와 같은 다른 센서를 사용하여 초기 변환에 대한 추가 정보를 얻는 것이다[59]. 또 다른 방법은 카메라가 포착한 시각적 정보에만 의존하는 것이다. 예를 들어, [60]에서 저자들은 비전 데이터, 카메라의 장착 지점 및 도로 표면의 평면을 사용하여 번역에서 스케일 문제를 해결하는 알고리즘을 제안했다. 이 방법은 척도의 모호성을 지속적으로 해결하고 척도 드리프트를 줄일 수 있다. 또한, [61]에서는 심층 컨볼루션 신경망(CNN) 알고리즘을 이용한 단안 카메라의 시각 데이터만을 이용한 스케일 복구 알고리즘이 제안되었다. 이 방법의 장점은 척도를 복구하고 [60]에서와 같이 고정된 기준면에서가 아니라 전체 환경의 구조에서 척도 표류를 제거할 수 있다는 것이다. 그러나 CNN의 많은 계산 요구로 인해 실제 응용 프로그램에서는 불가능하다.

2.3.3. 관성 비주얼 오도메트리

관성 센서는 드리프트 문제가 발생한다는 단점이 있지만, 업데이트 주기가 1kHz 수준으로 상당히 짧다. 반면 비주얼 오도메트리로 위치를 정확히 업데이트하더라도 자동차가 방향을 급격히 틀면 상대적으로 느린 이미지 업데이트 속도로 인해 일치하는 특징점이 줄어들어 현재 위치를 놓치게 된다. 영상 센서에서 측정된 값과 관성 센서로 측정된 값은 상호 보완 관계에 있기 때문에 두 센서를 결합하면 자율 주행 시스템에서 요구하는 수준의 견고하고 정확한 위치 추정 및 매핑 결과를 얻을 수 있다. 이렇게 영상 센서와 관성 센서를 결합해 실시간으로 위치 추정을 수행하는 기술을 관성 비주얼 오도메트리(Visual Inertial Odometry, VIO)라 부른다. 관성 비주얼 오도메트리에서 영상-관성 추정 문제를 해결하는 방법은 크게 두 가지 배치 비선형 최적화 방법(batch nonlinear optimization method)[62]과 재귀 필터링 방법(recursive filtering method)[59]이 있다. 배치 비선형 최적화 방법은 관성 센서로 인한 오차와 영상 센서로 인한 시각 재투영 오차 모두를 최소화하는 방법이고 재귀 필터링 방법은 관성 센서의 측정치를 상태 전파에 사용하고 영상 관측치를 상태 업데이트에 사용하는 최적화 방법이다.

[62]에서는 배치 영상 SLAM에 관성 센서 측정 값을 포함시키는 방식을 제안했다. 이 방식에서는 카메라의 측정 시점에서 관측한 랜드마크의 재투영 오차를 최소화하는 비선형 최적화 시스템을 구현해 카메라와 랜드마크의 위치를 추정했다.



관성 센서를 적용하면 연속적인 자세 사이의 시간적 제약이 발생할 뿐만 아니라, 연속적인 속도 및 로봇의 상태 벡터가 증강된 가속도계 및 각속도계에 대한 관성 센서의 바이어스 추정치에서도 시간적인 제약이 발생한다. 이 논문의 저자는 영상-관성 위치 추정과 매핑 문제를 영상 재투영 오차와 관성 센서의 시간 오차를 합친 비용 함수 $J(x)$ 에 대한 최적화 문제로 만들었다. 여기서 x 는 자동차의 현재 상태를 표현 한다.

이 알고리즘의 첫 단계는 전파(propagation)다. 처음에는 관성 측정치로 자동차의 자세를 전파해서 자세에 대한 불확실한 추정치를 미리 구한다. 이때 랜드마크에 대해 충분히 잘 알려진 3D 위치가 포함된 로컬 지도와 과거의 프레임을 사용할 수 있다고 가정한다. 랜드마크의 연관성을 추정하는 첫 단계로 3차원-2차원 매칭을 수행한다. 그런 다음, 2차원-2차원 매칭을 통해 3차원 랜드마크와는 관계없는 핵심적인 연관성을 추정한다. 이후 3D 랜드마크에 대한 새로운 위치를 초기화하기 위해 삼각 측위를 수행한다. 이때 현재 프레임과 사용 가능한 이전 프레임 사이의 스테레오 이미지 쌍에 대한 스테레오 삼각 측위가 수행 된다.

두 번째 단계는 최적화다. 새로운 이미지 프레임이 들어오면 이 이미지에서 특징을 추출해 새로운 3차원 점으로 확장하는데 이 점은 앞에서 설명했듯이 지도를 확장하는 데 활용한다. 이따금 프레임 하나를 키 프레임(key frame)으로 선택해 최적화를 구동시킨다. 키 프레임을 선정하기 위한 한 가지 간단한 방법은 매칭된 특징점에 비해 새로 검출된 특징점이 적은 프레임을 키 프레임으로 선정한다. 이렇게 새로 추가된 키 프레임은 기존에 있던 다른 모든 키 프레임과 함께 영상 재투영 오차와 관성 센서 오차를 줄이기 위한 전역 최적화에 사용된다. 실험 결과에 따르면 이 최적화 기법을 적용해 500m를 주행했을 때 0.3% 미만의 주행 오차가 발생했다.

최적화 방법을 통해 위치 추정 정확도를 높일 수 있지만, 알고리즘을 반복적으로 수행하기 때문에 연산 비용도 높아진다. 연산량을 줄이기 위해 EKF 기반의 위치 추정 기법이 제안됐다[59]. 이 논문은 EKF 알고리즘으로 카메라의 자세를 상태 벡터로 삼아 카메라의 슬라이딩 윈도우를 유지하도록 하고, 관측된 특징을 자세 사이의 확률적 제약 조건을 적용하는 데 사용했다. 이 알고리즘은 크게 세 단계로 진행된다. 첫 단계인 전파(propagation) 단계는 각 관성 측정치마다 필터의 상태와 공분산(covariance)을 전파한다. 두 번째 영상 등록(image register) 단계는 이미지가 기록될 때마다 현재 카메라 위치의 추정치로 상태 및 공분산 행렬을 보정한다. 세 번째 업데이트(update) 단계는 이미지에 대한 특징이 추출됐을 때 EKF 업데이트를 수행한다. 이 방식을 적용해 3.2km를 주행한 결과 최종 위치 오차는 주행 거리의 0.31 %에 해당하는 10m에 불과했다.



2.4. 센서 융합

지금까지 다양한 위치 추정 기법을 살펴보았다. 실전에서는 견고성과 안정성을 높이기 위해 다양한 센서를 조합해 위치 추정을 수행하는 방식을 많이 사용한다[63, 64, 65]. 이 절에서는 자율 주행 자동차에서 이렇게 센서 융합 기법을 적용하는 세 가지 사례를 소개하고 각 사례에서 사용하는 위치 추정 방식을 살펴 본다.

2.4.1. 어반 챌린지의 CMU 보스

그림 2.11에 나온 보스(BOSS)는 GPS, 라이다, 레이더, 카메라 등 다양한 센서를 탑재해 다른 자동차를 추적하고 장애물을 탐지하며, 도로 모델을 기준으로 위치 추정을 수행한다[63]. 또한 보스는 실제 도로에서 시속 48km의 속도로 안전하게 주행할 수 있다. 이 시스템은 처음부터 DARPA의 어반 챌린지(Urban Challenge)에 맞게 개발됐다. 그림 보스의 위치 추정 시스템에 대해 살펴보자.



그림 2.11. 카네기 멜론 대학(CMU)의 Boss[66]

보스의 위치 추정은 DGPS(differential GPS) 기반의 자세 추정부터 시작한다. 자세 추정 값은 상용 위치 추정 시스템을 통해 얻은 데이터와 차선의 랜드마크에 대한 측정치를 결합하는 방식으로 구한다. 이때 세부 정보가 있는 도로 지도를 활용해 도로 차선 표식을 측정한다. 글로벌 위치의 초깃값은 DGPS와 IMU 센서, 그리고 100Hz로 위치 측정치를 제공하는 휠 인코더를 조합한 센서 융합 시스템으로 추정하기 때문에 GPS 오류에 영향을 적게 받는다. GPS 신호가 안정적이라면 위치 추정 오차는 0.3m 이내로 줄일 수 있다. 또한 여러 센서를 사용하기 때문에 GPS 신호 없이 1km를 주행해도 위치 추정 오차가 1m 이내다.



이 시스템은 매우 정확하지만 차선에 대한 정보는 제공하지 않는다. 차선의 범위 탐지는 도로를 바라보도록 SICK LMS 라이다로 처리한다. 이 센서로 차선에 대한 표식을 탐지하는 방식으로 차선 범위를 가늠한다. 차선 표식은 도로의 다른 부분보다 밝기 때문에, 도로 전체에 대해 측정된 조명도에 기울기 함수를 적용하면 차선을 쉽게 탐지할 수 있다. 관측 결과에 나온 봉우리와 골짜기는 차선 표식의 경계일 가능성이 높다. 차선이 아닌 것을 차선으로 인식할(거짓 긍정 false positive) 가능성을 줄이기 위해, 간격이 적절한 봉우리와 골짜기 쌍만 차선 표식으로 간주한다. 이렇게 선정된 차선 표식 후보를 주변 영역의 조명을 기준으로 한 밝기에 따라 다시 걸러낸다. 이 과정을 거쳐 최종적으로 차선 표식의 위치를 결정한다.

보스는 위치 추정의 정확도를 더욱 높이기 위해 고정된 장애물과 차선에 대한 정보를 담은 도로 지도를 만들었다. 이 지도는 HD 지도의 원조인 썸이다. 이 지도는 기본적으로 센티미터급의 정확도를 갖는 디지털 지도를 기반으로, 라이다로 탐지한 기하학적 요소 정보를 추가했다. 기하학적 요소에는 차선 표식과 장애물이 있다. 보스의 맵핑 시스템은 자동차에 달린 여러 레이저 스캐너로 측정된 데이터를 취합해 측정 순간에 대한 지도와 시간적으로 보정된 장애물 지도를 만든다. 측정 순간에 대한 장애물 지도는 장애물이 움직이는지 검증하는 데 사용하고, 시간적으로 보정된 장애물 지도는 움직이는 장애물을 데이터에서 제거하는 데 사용되며, 지도에 나타나는 가짜 장애물의 숫자를 줄이도록 보정한다. 도로의 경계석, 수풀, 갓길 같은 기하학적 특징은 도심과 비포장 환경에서 길의 모양을 결정할 수 있는 정보를 제공한다. 라이다 데이터가 정밀하면 이러한 기하학적 특징을 먼 거리에서도 정확하게 감지하는 데 충분한 정보를 제공한다. 이러한 특징을 탐지하는 알고리즘은 다양한 형태의 도로 표지석 갓길 배수로 경사면의 변화에 충분히 대응할 수 있도록 견고하게 만들어야 한다. 예를 들어 보스 제작팀은 도로 분리대를 탐지하기 위해 두 가지 원칙을 적용해 라이다의 탐지 과정을 단순화했다. 첫째 도로 표면은 상대적으로 평평하고, 표면의 특성이 매우 느리게 변하며, 눈에 띄는 높이 변화는 도로의 가장자리에서 발생한다고 가정했다. 즉, 도로의 가장자리에 대한 주요한 특성을 지표면의 높이 차이로 간략화했다. 둘째 3차원 탐지군(point cloud)을 형성하지 않고, 각 라이다 스캔을 독립적으로 수행했다. 그래서 입력 데이터를 한 차원에 대해서만 처리하도록 알고리즘을 단순화했다. 이렇게 만든 도로 지도를 기반으로 파티클 필터를 적용해 자동차의 위치 추정을 수행한다.

2.4.2. 어반 챌린지의 스탠퍼드 주니어

그림 2.12에 나온 스탠퍼드 주니어(Stanford Junior(이하 주니어))는 어반 챌린지



에 참가한 스탠퍼드 대학교의 작품이다. 주니어는 2006년식 폭스바겐 파사트 왜건에 5개의 라이더, DGPS 보정 관성 항법 시스템, 5개의 레이더, 2개의 인텔 쿼드코어 컴퓨터를 탑재했다[64]. 이 자동차는 장애물 탐지 범위가 120m에 달하며, 어반 챌린지 규정에 맞추어 시속 48km까지 속도를 낼 수 있다.

CMU 보스와 마찬가지로 주니어도 실시간으로 GPS 좌표를 통합해주는 DGPS 보정 관성 항법 시스템 관성 측정치 휠 오도메트리 측정치를 이용해 위치 추정을 수행한다. 이 시스템의 실시간 위치 및 방향 오차는 대략 100cm/0.1° 이하다. 자동차의 상부에 설치된 라이더 센서는 실시간으로 인접한 도로를 3차원 구조로 측정한다. 라이더 센서는 차선 표식 탐지와 정밀한 위치 추정을 위한 도로의 적외선 반사 측정치도 제공한다.



그림 2.12. 스탠퍼드 대학교의 Junior[64]

주니어는 도로망에 대한 디지털 지도를 탑재하고 있다. 센서 피드백 없이 이러한 디지털 지도와 GPS 기반 관성 항법 시스템만으로는 차선을 유지할 수 있을 정도로 정확한 자동차 좌표를 보정할 수 없다. 게다가 디지털 지도가 완벽하지 않아서 다른 정보 없이 주행할수록 오차가 늘어난다. 따라서 주니어는 디지털 지도와 함께, 로컬 라이더 센서 측정 값으로 세밀한 위치 추정을 수행한다. 위치 추정을 수행할 때 도로 분리대 같은 장애물과 도로 반사라는 두 가지 정보를 활용한다. 도로 반사는 RIEGL LMS-QJ20과 SICK LMS 센서로 측정하며, 두 센서 모두 지표면을 향해 설치된다. 세밀한 위치 추정을 수행하기 위해 1차원 히스토그램 필터를 활용해 디지털 지도를 기준으로 자동차의 측방향 편차를 추정한다. 이 필터는 도로의 반사 정보와 도로 분리대의 발견 정보를 통해 측방향 편차에 대한 사후 확률 분포를 추정한다. 반사 패턴이 디지털 지도에 나온 차선 표식이나 도로 경계와 나란하면 두 간격만큼 확률적으로 보상한다. 반대로 관측된 도로 표지석이 지도에 나온 주행 경로와 부딪히면 둘 사이의 편차만큼 뺀다. 이러한 필터를 통해 GPS



기반 INS 시스템의 측정 위치에 대한 편차를 항상 정밀하게 추정할 수 있다.

2.4.3. 메르세데스 벤츠의 버사

앞 절에서 소개한 CMU 보스와 스탠퍼드 대학교의 주니어는 GNSS/INS 시스템으로, 먼저 개략적으로 위치 추정한 뒤에 라이다 스캔 데이터와 디지털 지도를 활용해 세밀한 위치 추정 결과를 구했다. 그런데 라이다 센서 기반의 방법에는 두 가지 문제가 있다. 첫째 라이다 장비 한 대 가격은 미화 80,000달러 이상으로 상당히 비싸다. 둘째 HD 지도를 만드는 데 라이다 가격보다 훨씬 많은 비용이 든다(한 도시의 HD 지도를 구축하고 유지하는데 수백만 달러가 든다). 이에 대한 대안으로 디지털 지도와 컴퓨터 비전을 활용하는 방법이 있다. 대표적인 예로 메르세데스 벤츠(Mercedes Benz)가 제작한 자율 주행 자동차인 버사(Bertha)가 있다. 버사는 정밀한 디지털 지도에 비전과 레이더 센서만으로 복잡한 교통 상황을 파악한다[65]. 이 절에서는 자율 주행 자동차 버사의 위치 추정 방식에 대해 자세히 살펴보자.

그림 2.13을 보면 버사 시스템은 다음과 같은 센서를 사용한다.



그림 2.13. 메르세데스 벤츠 자율 주행 자동차 버사[65]



먼저 GPS 모듈로 기본적인 위치 추정을 수행한다. 또한 120° 범위의 단거리 레이더 4개로 교차로의 상황을 감시한다. 그리고 자동차의 양측면에 장착된 2개의 장거리 레이더로 외곽 도로의 교차로에서 발생하는 빠른 교통 상황을 감시한다. 베이스라인이 35cm인 스테레오 카메라를 하나 달아서 60m 범위로 깊이 정보를 구한다. 대시보드 위에 광각 모노컬러 컬러 카메라를 설치해서 방향을 전환하는 중에도 신호등과 보행자를 인식한다. 후방에 장착된 1개의 광각 카메라는 자체 위치 추정에 활용한다.

또 다른 중요한 정보로 상세 디지털 지도가 있는데 여기에는 차선의 위치 같은 기하학적 정보뿐만 아니라 제한 속도와 신호등을 비롯한 교통 규칙에 관련된 정보도 담겨 있다. 버사에서 사용하는 디지털 지도는 스테레오 카메라로 찍은 영상을 기반으로 만들어졌다. 각 스테레오 이미지 쌍마다 자동차 주변의 환경에 대한 3D 모델과 세부적인 편차 이미지가 생성된다. 이렇게 만든 3D 포인트는 현실에 맞춰(world plane) 투영된 형태로 변환되며 기준 궤적에 대해 적층된다. 특징점 지도와 차선 표식을 포함한 지도를 추출하는 데는 일관성을 보장하기 위해 동일한 스테레오 이미지를 사용한다. 기준 궤적(reference trajectory)은 RTK GNSS/INS 항법 시스템으로 기록하는데, 자율 주행 과정에서 실시간으로 위치 추정을 수행할 때는 이렇게까지 복잡한 시스템을 사용하지 않는다. 버사 팀은 OpenStreetMap 프로젝트에 사용된 도구를 활용해 자율 주행에 쓰이는 지도를 만들고 관리한다[43].

버사를 위치 추정하려면 GNSS/INS 시스템의 위치 추정 오차가 반경 1m를 넘지 않아야 한다. 우선 위치 추정의 정확도를 센티미터급으로 높이기 위해 상호 보완 관계에 있는 2개의 지도 기반 위치 추정 알고리즘을 개발했다. 첫 번째 알고리즘은 자동차 바로 앞에 있는 랜드마크를 점 형태로 탐지하는 랜드마크 기반 위치 추정으로서 인공 구조물이 많은 도심지에 특히 효과적이며, 특정 기반 위치 추정이라고도 부른다. 두 번째 알고리즘은 외곽 지역에서 쉽게 감지할 수 있는 차선 표식과 도로 분리대를 활용하며 이를 관측한 결과를 지도 기반 위치 추정 값으로 변환하는 방식으로 차선 표식 기반 위치 추정이라고 부른다. 여기서 주목할 점은 두 시스템 모두 비주얼 주행거리계 기술을 활용한다는 점이다.

랜드마크 기반 위치 추정은 먼저 매핑 단계에서 스테레오 이미지를 가져온다. 그런 다음, 시험 자율 주행 과정에서 후방을 향해 설치된 모노컬러 카메라에서 또 다른 이미지를 가져온다. 두 이미지는 공간적으로 배치해서 설명자 기반 지점 특징 결합(descriptor-based point feature association)의 수단으로 사용한다. 지도 시퀀스에서의 두드러진 특징은 자동차 후방을 향해 설치된 카메라가 획득한 이미지에서 탐지된 특징과 연결한다. 스테레오 지도 시퀀스를 통해 랜드마크의 3차원 이미지를 재구성하고, 서로 일치하는 연관 특징을 통해 두 카메라 사이의 자세 변화



를 6차원 강체 변환 형태로 구할 수 있다. 이러한 6차원 변환결과를 지도 이미지의 글로벌 좌표에 대한 자세와 자동차에 장비한 휠 인코더 및 각속도 센서로부터 획득한 움직임 정보와 결합하면 글로벌 좌표에 대한 위치의 추정 값을 정확히 구할 수 있다.

외곽 도로 환경에서 고정된 특징은 차선 표식뿐일 때가 많다. 이로 인해 차선 표식 기반 위치 추정 시스템이 개발됐다. 차선 표식 기반 위치 추정의 첫 번째 단계는 모든 시각적 표식을 담은 정밀 지도의 제작이다. 도로 표식과 정지선뿐만 아니라, 도로분리대와 트램(tram)의 선로도 지도에 표시한다. 다음 단계인 실시간 위치 추정에서는 지도의 일부 영역을 현재 이미지에 투영한다. 이때 표본 지도에 대한 최근접 탐색법을 통해 이미지를 매칭한다. 이 과정에서 나온 잔차(residual)는 칼만 필터를 통해 최소화한다.

3. 자율 주행을 위한 인지

3.1. 개요

사람의 두뇌는 우리 주변의 환경을 인지하는 능력이 매우 뛰어나다. 심지어 너 무나도 좋은 성능을 당연하게 여기기도 한다. 도로를 건널 때 우리는 다가오는 차량이 있는지 확인한다. 우리를 향해 다가오고 있는 차가 있다면 다가오는 차량의 속도를 빠르게 예상할 수 있고, 도로를 건너는 것이 안전한지 판단할 수 있다. 어떤 장면을 볼 때, 우리의 눈은 그저 빛과 색의 패턴을 기록해서 두뇌로 전달한다. 두뇌는 전달받은 패턴을 과거의 경험을 이용해 실제 이미지로 해석한다. 실제로 우리는 눈의 광학적인 원리 때문에 위 아래가 뒤집어진 이미지를 보지만, 두뇌가 이미지를 올바르게 해석할 수 있어 공을 잡는 것과 같은 묘기를 부릴 수 있는 것이다. 하지만 자율 주행차가 이런 능력을 따라하는 것은 어렵다.

자율 주행차는 다양한 하드웨어 센서에 의존한다. 하지만 이런 센서가 생성한 미가공 데이터는 우리의 눈과 마찬가지로 본질적으로 무의미하다. 이런 데이터를 해석하고 해석한 데이터를 이용해 차량 주변의 환경을 그려나가는 것은 소프트웨어가 수행하는 일이다. 구체적으로 말하자면, 이런 해석을 수행하는 것이 바로 인지 소프트웨어이다. 1.5미터 높이의 가느다란 물체가 눈앞의 도로를 천천히 건너는 것을 봤을 때, 인지 소프트웨어는 이 물체가 보행자임을 인식하고 이 정보를 제어 소프트웨어로 전달할 수 있다. 이 소프트웨어는 차량이 보행자를 피할지 말지도 판단할 수 있다.

인지 기능의 목표는 차량 주변의 환경을 가능한 한 완전하고 정확하게 이해하는 것이다. 인지 기능은 이어지는 항법 기능의 의사 결정을 위한 근거가 된다. 인지에는 ‘나는 어디에 있는가?’와 ‘내 주변에는 무엇이 있는가?’에 대한 답변이 포함된다. 믿을 수 있는 인지 기능은 자율 주행차의 원활하고 안전한 작동을 보장하는 데 필수적이다. 이 장에서는 자율 주행차가 주변 환경을 어떻게 인지하는지를 자세히 다룰 것이다.

3.2. 데이터 세트

충분한 수의 샘플을 제공하는 데이터 세트는 알고리즘을 신속하게 반복할 수 있게 하여 성능의 정량적인 평가, 잠재적인 약점의 발견, 객관적인 비교를 할 수 있게 해준다. 컴퓨터 비전에서는 이미지 분류[67, 68], 의미 분할(semantic segmentation)[67-69], 옵티컬 플로우(optical flow)[70, 71], 스테레오 컴퓨터 비전[72, 73], 영상추적[74, 75] 같은 기본적인 문제에 대한 데이터 세트가 존재한다.

이러한 데이터 세트는 클라우드 소싱(crowd-sourcing)[67, 69]이나 기술적 접근법을 이용하는 등[70-73] 다양한 방법을 통해 수집된다. 각 항목의 데이터 세트에는 수 장에서부터 수백만 개의 분류된 데이터까지 다양한 학습용 표본이 있다. 일반적으로 데이터 세트의 크기가 크고 사실적인 이미지가 많으면 많을수록 실제 상황에 대해 알고리즘이 적은 편차와 높은 신뢰도의 성능을 보인다.

자율 주행의 경우, KITTI[76]나 Cityscapes[77] 같은 자율 주행에 특화된 데이터 세트가 존재한다. 이 데이터 세트는 자율주행 차량이 겪는 현실적인 상황에 대한 장면을 다양한 센서로 측정하는 방식을 통해 구현된다. 허나 이들은 평가에만 사용할 수 있기 때문에, 상용화를 위한 자율 주행 기술을 개발하는 사람은 현실에서 겪을 수 있는 광대한 상황을 충분히 다룰 수 있도록 큰 규모를 가질 뿐만 아니라 지속적으로 성장하는 데이터 세트를 만들어야 한다.

KITTI 데이터 세트는 카를스루에 공과대학(Karlsruhe Institute of Technology, KIT)과 시카고 토요타 기술 연구소(Toyota Technological Institute at Chicago, TTIC)가 2012년에 합작해 만든 프로젝트다. KITTI는 전용 웹사이트 (<http://www.cvlibs.net/datasets/kitti/>)를 보유하고 있다. 이 프로젝트의 목적은 자율 주행을 위한 현실적이고 높은 난이도의 데이터 세트를 만드는 데 있다. 데이터 세트에 필요한 원시 데이터는 그림 3.1에 묘사된 형태의 차량을 통해 수집했다.



그림 3.1. KITTI 자동차의 사진 [78]

이 자동차에는 다음과 같은 장비가 장치되어 있다.

- 1개의 3차원 라이다: Velodyne HDL-64E
- 1.4메가픽셀 해상도의 사진을 10Hz 주기로 찍을 수 있는 2개의 흑백 카메라: Point Grey Flea 2 (FL2-14S3M-C)



- 1.4메가픽셀 해상도의 사진을 10Hz 주기로 찍을 수 있는 2개의 컬러 카메라: Point Grey Flea 2 (FL2-14S3C-C)
- 1개의 6축 관성 항법 시스템(GPS/IMU): OXTS RT 3003

KITTI 데이터 세트는 다음의 요소로 구성되어 있다.

- 스테레오 데이터와 옵티컬 플로우 데이터: 두 카메라를 동시에 사용함으로써 하나의 스테레오 이미지 쌍을 얻을 수 있다. 옵티컬 플로우 이미지 쌍은 한 카메라로 연속적인 촬영을 수행함으로써 획득한다. 이 데이터에는 총 200개의 학습용 장면과 200개의 시험용 장면이 존재하며, 각 장면당 4개의 컬러 이미지가 무손실 png 형식으로 저장되어 있다. 실제 변위 데이터는 대략 50%의 픽셀에 포함되어 있다.
- 시각적 주행거리 측정 데이터: 39.2km의 거리를 40,000개 이상의 프레임으로 구성한 22개의 스테레오 이미지 시퀀스가 존재하며, 무손실 png 파일로 저장된다.
- 객체 탐지 데이터: 객체의 크기와 방향을 알 수 있도록 규칙적으로 표시된 2차원 및 3차원 데이터다. 2차원 데이터는 총 80,256개의 레이블이 지정된 객체로 구성된 7481개의 훈련 영상과 7518개의 테스트 영상으로 구성된다. 모든 이미지는 컬러이며 png로 저장된다. 그림 3.2에 나타난 것과 같이 세단, 밴, 트럭, 보행자, 자전거 탑승자 등의 객체가 있다. 객체가 가려진 경우도 있으며, 보통 각 이미지에 여러 개의 객체가 나타나 있다.



그림 3.2. 2차원 객체 탐지 데이터 [76]

3차원 객체 탐지 데이터는 총 80,256개의 레이블이 지정된 객체로 구성된 해당 포인트 클라우드뿐만 아니라 7481개의 훈련 영상과 7518개의 테스트 영상으로 구성된다. 그림 3.3에 나타난 것과 같이 세단, 밴, 보행자 등의 객체가 있다.

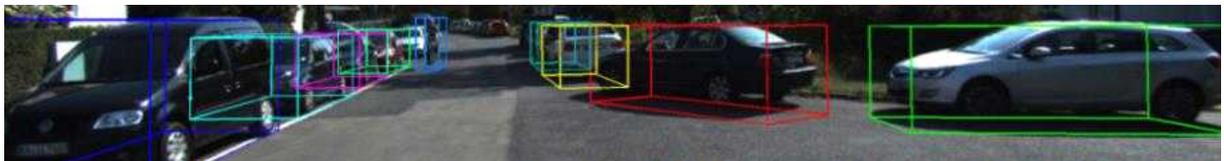


그림 3.3. 3차원 객체 탐지 데이터 [76]

- 객체 추적 데이터: 21개의 학습용 이미지 시퀀스와 29개의 시험용 이미지 시퀀스가 존재한다. 그림 3.4에 나타난 것과 같이 주로 추적하는 객체는 보행자와



차량이다.



그림 3.4. 객체 추적 데이터[76]

• 객체 추적 및 분할(MOTS, Multi-Object Tracking and Segmentation): 다중 객체 및 분할(MOTS) 데이터는 21개의 훈련 시퀀스와 29개의 테스트 시퀀스로 구성된다. KITTI의 객체 추적 데이터를 기반으로 하며 주석을 다중 객체 및 분할(MOTS) 작업으로 확장했다. 이를 위해 모든 객체에 대해 고밀도 픽셀 단위 분할 레이블을 추가했다. 그림 3.5와 같이 주로 추적 및 분할하는 객체는 보행자와 차량이다.



그림 3.5. 객체 추적 및 분할 데이터[76]

• 도로 분석 데이터: 노선 표시가 없는 시골의 도로 노선 표시가 있는 도시의 도로, 여러 개의 차선이 있는 도시의 도로 등 다양한 환경의 노면에 대해 대응할 수 있는 289개의 학습용 데이터와 290개의 시험용 데이터가 존재한다.

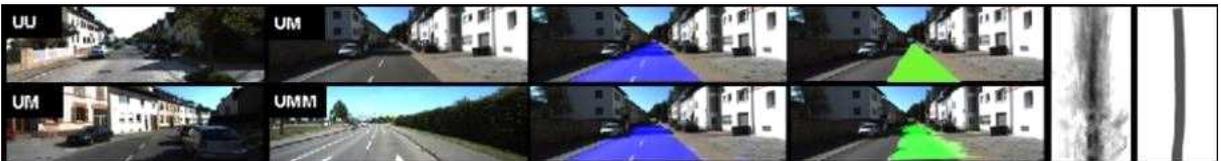


그림 3.6. 도로 분석 데이터[76]

KITTI와 Cityscapes 데이터 세트는 세 가지 측면에서 전통적인 컴퓨터 비전 데이터 세트와 차별화 된다.

- 다양한 센서와 3차원 라이다를 사용하기 때문에 정밀한 값을 얻을 수 있고, 이를 통해 고수준의 3차원 형상을 얻을 수 있다.
- 통제되거나 용도에 맞게 만들어진 실험실 환경과는 달리 실제 환경을 통해 데이터를 획득했다.
- 자율주행에서 필요한 다양한 인지 능력에 걸맞은 데이터를 포함한다.



이러한 특성 덕분에 KITTI와 Cityscapes 데이터 세트가 널리 사용됐고, 이 데이터 세트를 기반으로 하는 새로운 알고리즘의 제안 및 평가 역시 지속적으로 이뤄지고 있다.

3.3. 객체 탐지

자율 주행차에 필요한 기본적인 능력 중 하나는 객체 탐지다. 자율 주행차는 자동차나 보행자처럼 교통에 영향을 주는 요인들과 도로를 공유한다. 이 외에도 장애물이나 차선의 분리대 같은 다양한 객체가 도로에 존재한다. 따라서 충돌/사고를 예방하기 위해 이러한 객체들을 빠르고 정확하게 탐지하는 것은 안전상의 이유로도 매우 중요하다. 또한, 환경을 적절하게 이해해 현재 상황에서 가능한 최선의 선택을 할 수 있도록 하는 것 또한 중요하다. 운전하는 사람으로서, 우리는 때때로 무의식적인 광범위 동시 객체 탐지 작업을 수행한다. 우리는 자동차, 보행자, 자전거와 같이 환경에서 움직이는 객체뿐만 아니라 차로 범위, 교통 표지판, 신호등과 그 외의 많은 위험 구성 요소 같은 정적인 구성 요소들도 인식할 필요가 있다. 객체의 탐지는 컴퓨터 비전에서 기본적인 문제 중 하나이고, 이를 해결하기 위한 다양한 알고리즘이 제안되어 왔다. 하지만 오늘날에도 컴퓨터가 이런 능력을 복제하는 것은 매우 어려운 일이다. 그러나 딥러닝 같은 유망한 기술들이 격차를 빠르게 좁히고 있다.

컴퓨터 비전 연구에서 객체 탐지는 일반적으로 다음과 같은 부분 문제(subproblem)로 나뉜다.

- 객체에 대한 바운딩 박스(bounding box)의 결정
- 객체 분류, 즉 탐지한 객체를 사전에 정의한 클래스로 분류
- 의미 분할(semantic segmentation), 즉 이미지를 의미적으로 주요한 부분으로 분할하고 각 부분을 사전에 정의한 의미 영역 중 하나로 분류

그림 3.7은 위의 부분제들 사이의 차이점을 나타낸다.

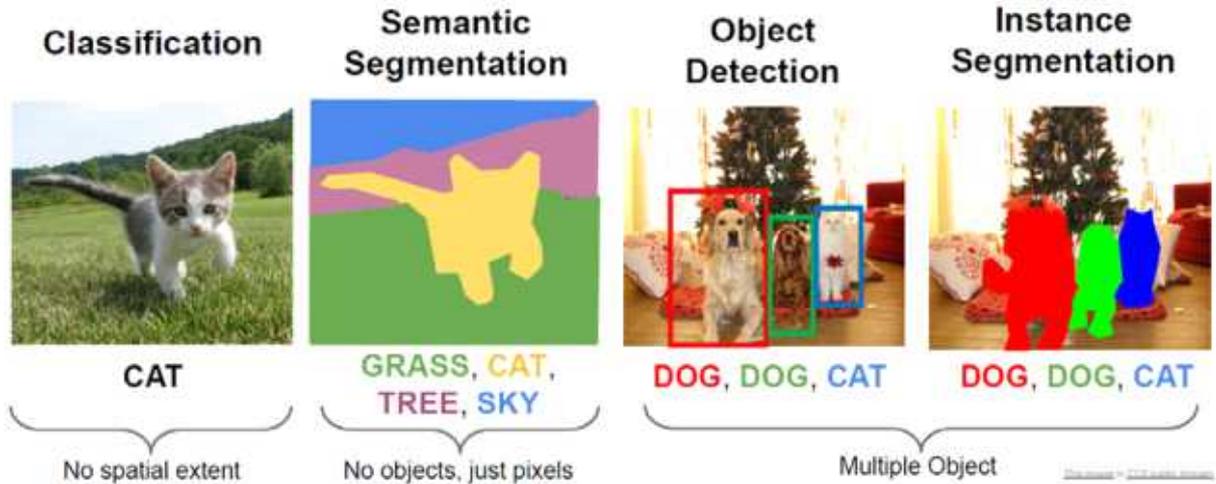


그림 3.7. 객체 탐지와 분류, 의미 분할 비교 (출처: 스탠퍼드 대학 cs231n-2019-Lectuer 12)

객체 탐지는 1960년대 중반부터 컴퓨터 과학 영역에서 활발히 연구되어 왔으며, 이 문제를 해결하기 위한 많은 기법이 제안됐다. [81]은 인기 있는 접근 방식들에 대한 연대기적 개요를 간략히 제시했다. 일반적으로, 객체 탐지 문제를 해결하는 것은 다음과 같은 단계를 포함한다(그림 3.8).

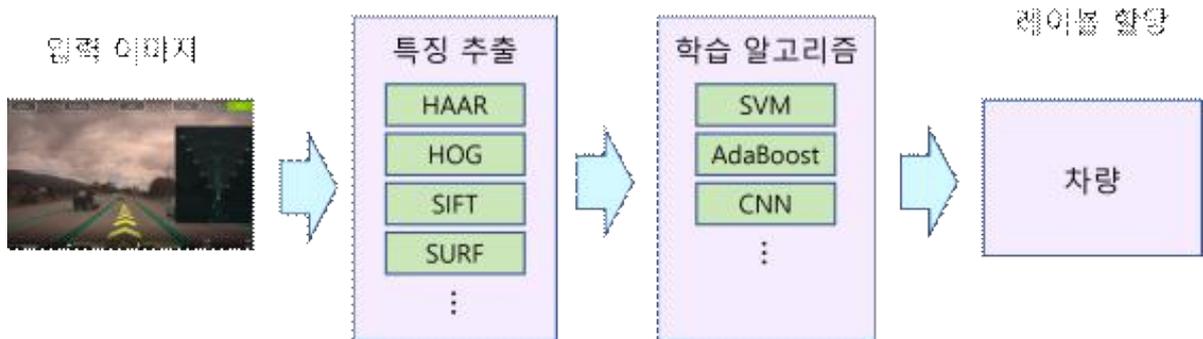


그림 3.8. 객체 탐지의 파이프 라인

전통적으로 객체 탐지의 파이프라인은 입력 이미지를 전처리하고 관심 영역을 탐지한 다음, 마지막으로 탐지된 객체를 분류하는 것으로 이뤄져 있다.

- 전처리

전처리 단계는 이미지를 ‘정규화’ 한다. 즉, 미가공(raw) 이미지에 대한 조정을 수행해 후속 특징 추출 단계에서 예측되는 입력과 매칭시킨다. 이 단계에는 이미지 회전 및 크기 조절, 강도 조절 등이 포함될 수 있다. 실제로 수행할 작업은 애플리케이션에 따라 다르다. 어떤 접근 방식은 심지어 전처리 단계를 완전히 뛰어넘기도 한다.



- 특징 추출

특징 추출 단계는 불필요하거나 관련 없는 정보를 이미지에서 제거하고 분류에 연관된 정보(혹은 특징)만 유지 시킨다. 특징 추출 단계를 통해 이미지가 특징 지도라 불리는 또 다른 형태로 변환 된다.

- 분류

마지막 단계는 특징 지도를 사전에 정의한 클래스를 나타내는 참조 특징 지도들과 각각 매칭시키는 것이다.

3.3.1. 특징 추출

특징 공학(feature engineering), 즉 각 클래스를 모든 클래스와 명확히 구별할 수 있도록 하는 특징 설명자(descriptor)의 설계는 객체 탐지에서 꽤 어려운 부분이다. 간단한 테이블 분류 체계를 만드는 것을 생각해보자. 컴퓨터에 테이블 이미지를 입력하면 ‘테이블’을 올바르게 출력하고, 그렇지 않으면 ‘테이블이 아님’을 출력하려고 한다. 간단한 특징 설명자는 ‘네 개의 다리를 갖는 테이블’이라고 명시한 함수일 것이다. 이런 특징 설명자를 기반으로 이미지에서 다리와 비슷한 파트에 관련된 정보만을 추출하고, 나머지 모든 정보는 폐기한다. 분류 단계에서는 다리와 비슷한 이미지 파트의 개수가 결정되고, 이 정보를 바탕으로 결론을 낸다. 이렇게 빈약한 특징 설명자를 사용한다면, 마지막 단계에서 수행하는 분류를 아무리 잘 수행하더라도 객체 분류 결과는 분명히 미흡할 것이다. 외다리 바 테이블처럼 네 개의 다리가 없는 테이블은 많다. 마찬가지로, 이 세상에는 개와 같이 네 개의 다리를 가졌지만, 테이블이 아닌 것들도 많다.

다행히 대규모 객체 탐지 문제를 해결하기 위해 다수의 일반적 특징 설명자 알고리즘이 정의됐고 성공적으로 적용되고 있다. 아래에서 설명할 알고리즘은 가장 인기 있는 알고리즘들 중 일부다.

3.3.1.1. HOG

HOG(Histogram of Oriented Gradients) 설명자[12]는 국부 객체의 생김새와 모양을 나타내기 위해 강도 구배나 에지 방향의 분포(혹은 히스토그램)를 사용한다. 그림 3.9(b)와 같이, 이 알고리즘은 이미지를 작은 셀로 나누고, 셀의 각 픽셀에 대한 구배의 방향 히스토그램을 계산한다. 일반적으로 국부 히스토그램의 대비는 모든 국부 히스토그램이 결합되어 최종 설명자를 형성하기 전에 블록이나 연결된 셀 집합에 걸친 평균 강도 값을 이용해 정규화된다.

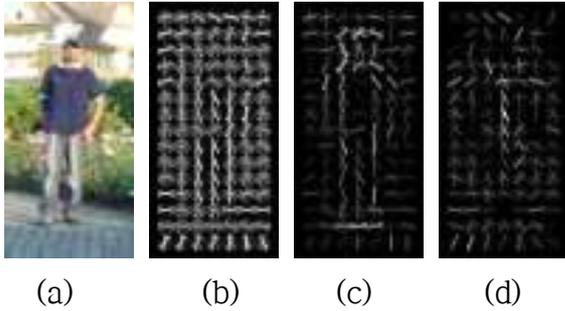


그림 3.9. (a) 테스트 이미지, (b) 계산된 HOG 설명자, (c, d) SVM 가중치를 적용한 HOG 계산자(출처: [79])

그림 3.9에서 이 알고리즘은 입력 이미지를 전처리하고, 슬라이딩 탐지 윈도우에 대한 HOG 특징을 계산한 다음 이들을 선형 SVM 분류기에 통과시켜 탐지를 수행한다. 이 알고리즘은 객체의 걸모습을 용도에 맞게 설계한 HOG 특징을 통해 획득하고, 비선형성이 강한 객체의 표현은 선형 SVM에 맡긴다.

HOG 알고리즘에 의해 연결된 객체들은 외관이 복잡해서 다루기가 어렵다. 이 문제는 펠젠스발프(Felzenszwalb) 등이 제안한 DPM(Deformable Part Model)을 통해 객체를 간단한 부품들로 나누어 쉬운 부품의 조합으로 강체가 아닌 객체들을 표현할 수 있다. 이 방법을 통해 전체 객체의 걸모습을 모델링하기 위한 학습용 예제의 숫자를 줄일 수 있다. DPM (그림 3.10)은 다양한 크기에 대한 객체의 가설 및 부품 배치의 제약에 대한 공간 조합 모델을 구축하기 위해 HOG 특정 피라미드를 사용하고, 부품의 위치처럼 드러나지 않은 변수를 처리하기 위해 잠재 SVM을 사용한다.

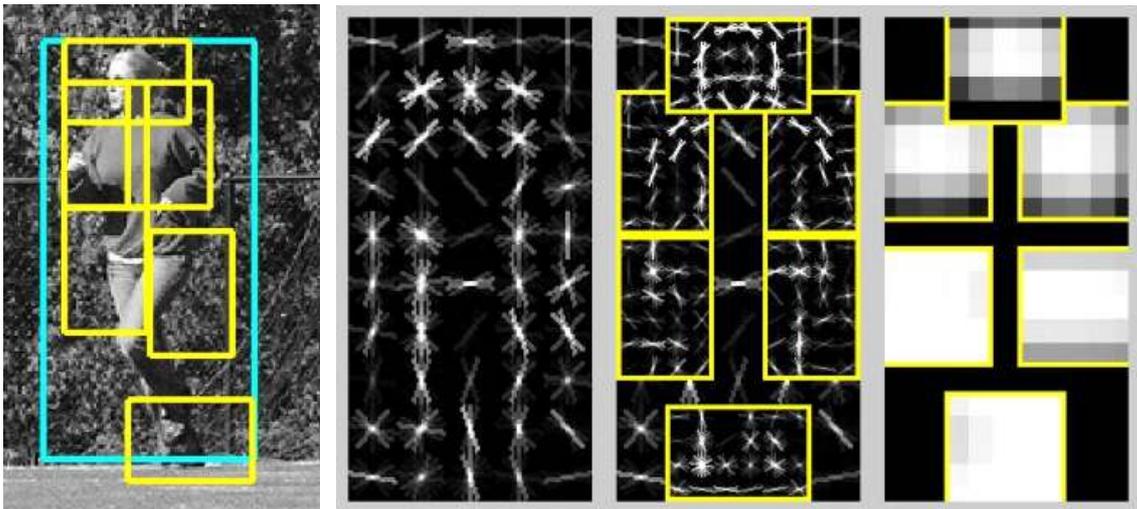


그림 3.10. 변형 부품 모델[80]



3.3.1.2. SIFT

SIFT(Scale-Invariant Feature Transform)[21]는 이미지를 불변 키포인트 집합, 즉 이미지에서 회전, 이동, 스케일, 조명과 기타 관측 조건에 따라 변하지 않는 국부 특징으로 나타낸다. 그림 3.11은 키포인트 선택 단계를 보여준다. SIFT에서 키포인트는 척도, 방향, 위치를 나타내는 벡터로 표시된다. 그림 3.11(b)는 초기 832개의 키포인트를 보여주고 있다. 초기 키포인트는 가우스 함수의 차이의 최대, 최소 위치에 위치한다. 그림 3.11(c)에서 임계값을 적용한 후 남은 729개의 키포인트를 보여준다. 그림 3.11(d)에서 주곡률 비율에 대한 추가 임계값을 적용하고 남은 536개의 키포인트를 보여준다.

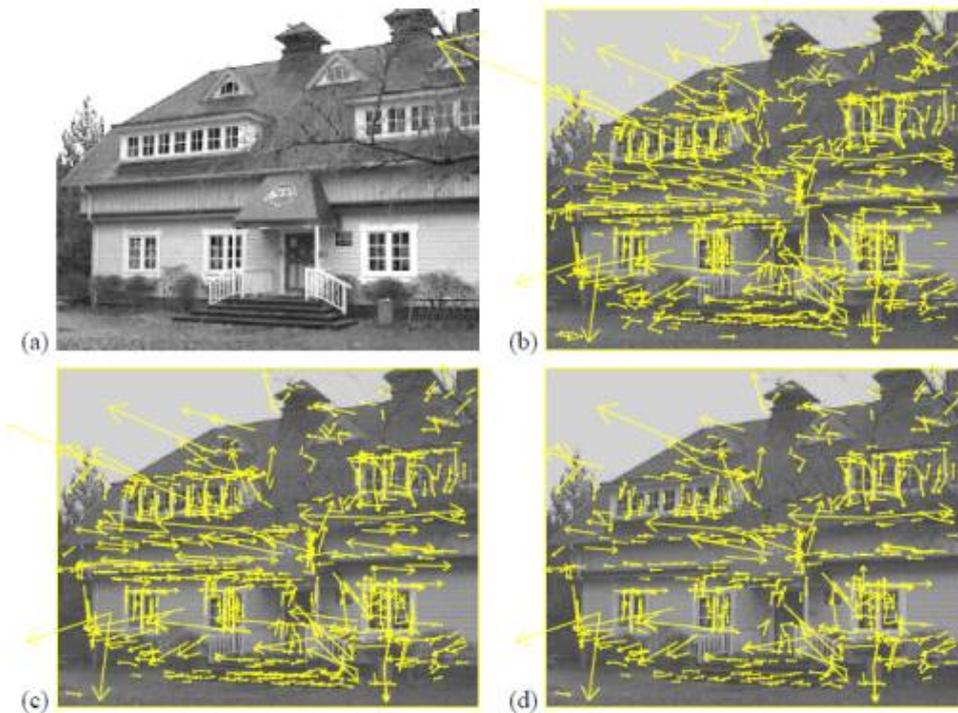


그림 3.11. SIFT에서의 키포인트 선택 단계(출처: [82])

3.3.1.3. MSER

MSER(Maximally Stable Extremal Regions)[83]은 주변 환경과의 상대적 영역 (혹은 연결 픽셀 집합) 변화 특성을 탐지하는 방식으로 동작하는 블롭(blob) 탐지 기법이다. 그림 3.12처럼 MSER은 이미지를 강도 변화에도 불구하고 최대한 안정적이거나 사실상 변하지 않는 영역의 집합으로 설명한다. 즉, 넓은 범위의 밝기에서도 볼 수 있는 영역을 찾는다는 것이다. MSER 영역은 일반적으로 실제 모양에 맞춰진 타원을 사용해 설명된다. MSER은 SIFT에 비해 빠르고 왜곡과 같은 아핀 변환(Affine transformation)에도 변하지 않는다[84].



그림 3.12. MSER의 적용 예(출처: [84])

3.3.2. 분류

객체 탐지 작업의 마지막 단계는 특징 추출 단계에서 추출한 특징을 ‘차량’, ‘보행자’, ‘트럭’ 등 사전에 정의한 클래스로 분류하는 것이다. 일반적으로 분류 작업은 인공지능 분류 알고리즘을 이용해 수행한다. 널리 쓰이는 분류 알고리즘에는 서포트 벡터 머신, 랜덤 포레스트, 인공 신경망이 있다.

3.3.2.1. 서포트 벡터 머신

서포트 벡터 머신(SVM, Support Vector Machine)[106]은 가장 인기 있고 효과적인 분류 알고리즘으로, 서로 다른 클래스 레이블 집합을 최적으로 분리하는 초평면(hyperplane)을 탐색한다. 단순한 선형 함수를 사용해 클래스를 분리하는 것은 보통 불가능하다. 그러나 고차원 공간에서는 분리할 수 없는 데이터를 선형 분리할 수 있으며 최적의 분리 초평면도 구할 수 있다. 따라서 입력 데이터는 그림 3.13과 같이 비선형 매핑(혹은 커널) 함수의 도움으로 고차원 특징 공간으로 변환되고 분리 초평면에 따라 분류된다.

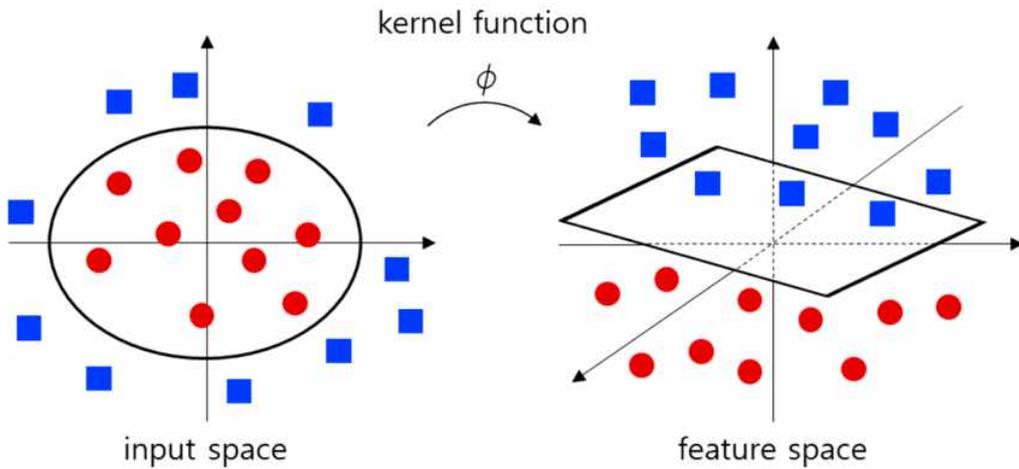


그림 3.13. SVM을 적용한 입력 공간에서 특징 공간으로의 매핑(출처: [85])

3.3.2.2. 랜덤 포레스트

그림 3.13과 같이, 랜덤 포레스트(Random Forest)[86]는 데이터 무작위 선택과 특징 서브셋(subset)을 통해 자동 생성되는 다중 결정 트리의 집합이다. 분류 결과는 다수결로 결정되는데 모든 결정 트리에서 가장 인기 있는 결과를 선택한다는 것이다. 단일 결정 트리와 비교할 때 랜덤 포레스트는 무작위 노이즈를 모델의 일부로 통합시키기 때문에 오버피팅(과대적합)에 더 견고하다. 랜덤 포레스트는 또한 트리의 평균 효과 덕분에 낮은 분산을 가진다[87].

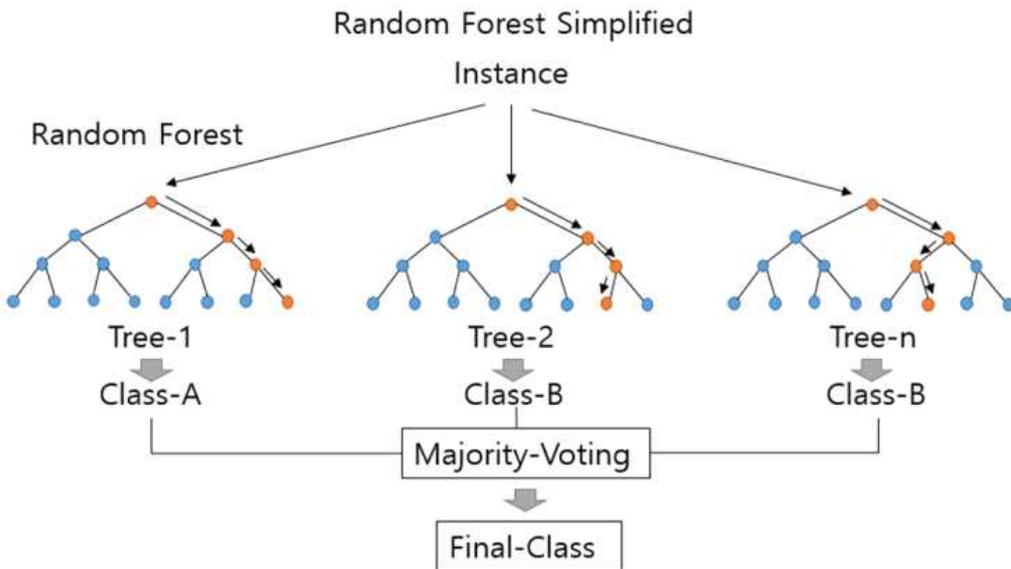


그림 3.14. 랜덤 포레스트의 구조(출처: [88])

3.3.2.3. 인공 신경망

인공 신경망(ANN, Artificial Neural Network)은 서로 연결된 노드(혹은 뉴런)의

다층 시스템이다. 일반적으로 비선형 분류에는 다층 퍼셉트론(MLP, Multi-Layer Perceptron)으로 알려진 ANN의 일종을 채택한다. MLP는 적어도 세 개의 층(입력, 은닉, 출력)으로 이뤄져 있고 입력층의 노드로 각 특징을 표현한다. MLP의 훈련에는 그림 3.15와 같이 정확한 분류를 할 수 있을 때까지 각 노드의 가중치를 전방과 후방 방향으로 반복 갱신하는 역전파(backpropagation) 알고리즘[89]을 사용한다. 전방 방향에서는 입력 데이터에 노드의 가중치를 곱해 구한 결과에 비선형 활성화 함수를 적용한 후 출력층에서 최종 결과를 얻을 때까지 다음 층으로 전파한다. 실제 결과와 예측 결과 사이의 오차를 측정하고, 이에 따라 출력층부터 입력층까지(즉, 역전파) 각 노드의 가중치를 조정해 오차를 줄인다.



그림 3.15. 오차 역전파의 개념(출처: <https://wikidocs.net/37406>)

자율 주행 차량은 가까운 거리에 보행자가 있는 교통 상황에서도 항법을 수행할 수 있어야 한다. 안전상의 이유로 보행자의 탐지는 특히 더 중요하다. 또한 사람의 행동은 종종 예측하기가 어려운데 사람의 걸모습이 매우 다양하고 신체의 일부가 가려질 수 있기 때문이다. 오늘날의 최신 보행자 탐지 기술에는 컨볼루션 인공신경망이 사용된다.

3.4. 분할

분할, 좀 더 정확하게 말하자면 실제 사례에도 적용할 수 있는 수준의 의미 분할은 자율 주행을 충분히 잘 수행하기 위해 필요한 객체 탐지를 자연스럽게 발전시킨 형태로 볼 수 있다. 카메라를 통해 획득한 이미지를 분석해 의미 있는 부분을 분할하는 것을 통해 자율 주행 차량이 주변 환경을 구조적으로 이해할 수 있다(그림 3.16).



그림 3.16. 취리히에서의 한 장면에 대한 의미 분할, Cilyscapes 데이터 세트[77]

전통적으로, 의미 분할은 정점이 픽셀 혹은 슈퍼 픽셀인 그래프의 라벨을 어떻게 잡을 것인지에 대한 문제가 된다. 이에 대해 CRF(conditional random field) 같은 그래픽 모델의 추론 알고리즘을 적용할 수 있다[90, 91]. 이 방식에서는 픽셀 혹은 슈퍼 픽셀을 의미하는 정점을 통해 CRF를 생성한다. CRF에서 각 노드의 라벨은 상응하는 이미지 위치에서 추출한 특징에 대해 사전 정의 세트로부터 가져올 수 있다. 노드 사이의 경계는 공간의 안정성이나 라벨의 상관관계 등에 대한 제약을 나타낸다 (그림 3.17).

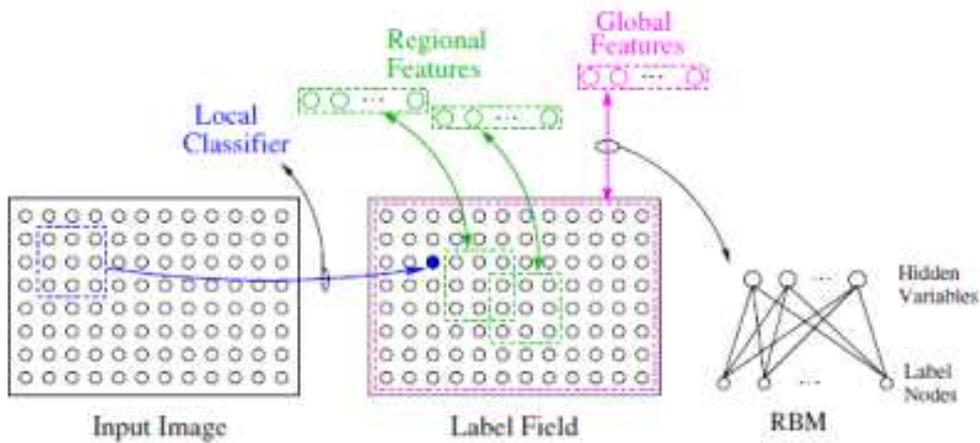


그림 3.17. 히(He) 등의 그래픽 모델[90]

CRF는 의미 분할 문제에 적절한 접근법이다. 그럼에도 불구하고 이미지의 차원, 입력되는 특징의 개수, 혹은 라벨의 종류가 많아지는 경우 처리 속도가 늦어지고, 이미지의 장거리 종속성 long range dependency을 획득하기 어려워진다. [92]에서 제안한 고효율 추론 알고리즘은 짝지을 수 있는 모든 픽셀 쌍에 대한 완전 연결형 CRF의 속도를 효과적으로 향상했고[93]에서 제안한 알고리즘은 객체 클래스



의 동시발생(co-occurrence) 또한 포함하게 했다. 의미 분할은 근본적으로 다중 이미지 특징과 맥락에 대한 추론을 통해 밀도가 높은 클래스의 라벨을 예측할 수 있어야 한다. 다음 장에서 딥러닝을 통해 의미 분할을 어떻게 다루는지에 대해 이야기 한다.

3.5. 다중 센서 데이터 융합

3.5.1. 스테레오와 깊이

자율 주행 차량은 3차원 공간에서 움직인다. 따라서 인지에 있어 깊이 같은 3차원 공간 정보를 생성하는 것은 필수적으로 인지해야 한다. 라이다는 매우 정밀한 깊이 데이터를 제공하지만, 3차원 공간을 낮은 밀도로 탐지한다는 한계가 있다. 하나의 이미지는 깊이 외에 색이나 질감 같은 정보를 공간적으로 밀도 있게 제공한다. 사람이 두 눈을 통해 3차원 공간을 시각적으로 인지할 수 있는 것과 마찬가지로, 스테레오 카메라를 약간 다른 각도에서 동시에 촬영함으로써 깊이 정보를 획득할 수 있다.

스테레오 카메라로부터 얻은 이미지 쌍(I_l , I_r)이 주어졌을 때, 스테레오 컴퓨터 비전은 근본적으로 좌측의 이미지 I_l 에서의 픽셀과 우측의 이미지 I_r 에서의 픽셀을 비용 함수에 기반해 어떻게 대응시킬 것인지에 대한 문제가 된다. 대응 픽셀들이 물리적으로 동일한 지점에 매핑된다고 가정하기 때문에 두 픽셀은 같은 모양을 갖는다.

특징을 기반으로 하는 방법은 픽셀의 값을 테두리나 모서리처럼 간단한 요소나 SIFT[82], SURF[94] 등의 정교한 알고리즘을 통해 생성한 값으로 대체해 변별력을 높인다. 이를 통해 신뢰도가 더 높은 매칭을 할 수 있지만, 공간적 관련성은 떨어진다.

대응 문제를 해결하는 또 다른 방법은 최적화다. 특징 기반 방법과 영역 기반 방법은 모두 지역적인 방법으로 볼 수 있는데 이는 로컬 정보를 바탕으로 d 를 계산하기 때문이다. 반면 글로벌 방법은 매칭을 외관이 변하지 않고 공간이 안정적이라는 가정하에서의 에너지 최소화로 접근한다. 글로벌 최적해를 구하기 위해 변분법(variational method), 다이내믹 프로그래밍(dynamic programming), 혹은 신뢰도 전파(Belief Propagation) 등의 방법을 사용할 수 있다.

SGM(Semi-Global Matching)[95]은 가장 널리 알려진 스테레오 매칭 알고리즘이다. SGM은 이론적으로 그 타당함이 입증됐고[96] 상당히 빠른 알고리즘으로 알려져 있다[97]. SGM은 각 픽셀의 1차원 선들을 따라 계산 에너지 함수와 안정성에 대한 에너지 함수를 사용하는 글로벌한 방법이다. 최근에는 딥러닝을 기반으로 하는 방법이 가장 좋은 성능을 보이고 있다.



3.5.2. 옵티컬 플로우

옵티컬 플로우[98]는 기본적인 컴퓨터 비전 분야 중 하나로, 두 이미지 사이의 강도에 대한 2차원 움직임으로 정의된다. 이때 2차원적 움직임은 물리 세계의 3차원적인 움직임과 관련이 있긴 하지만 두 움직임 사이의 차이가 존재한다. 옵티컬 플로우는 외관이 변하지 않는다는 것을 조건으로 한다.

옵티컬 플로우는 스테레오 컴퓨터 비전 문제보다 확실히 복잡하다. 스테레오 컴퓨터 비전에서는 이미지 쌍이 동시에 획득되고 둘 사이의 기하관계가 편차의 영향을 받으며, 대부분의 경우에서 외관의 불변이 인정된다. 옵티컬 플로우는 이미지 쌍이 약간 다른 시간대에서 획득되고 움직임은 조명 반사 투명도 같은 다양한 요소의 일부일 뿐이다. 외관의 불변성은 시시각각 위반된다. 옵티컬 플로우에서의 또 다른 문제는 조리개 문제 즉 공간 d 에서 한 조건과 다른 명확하지 않은 두 요소의 차이(그림 3.18)다. 이는 움직이는 공간 d 에 대해 안정성 제약 조건을 부여함으로써 해결할 수 있다.

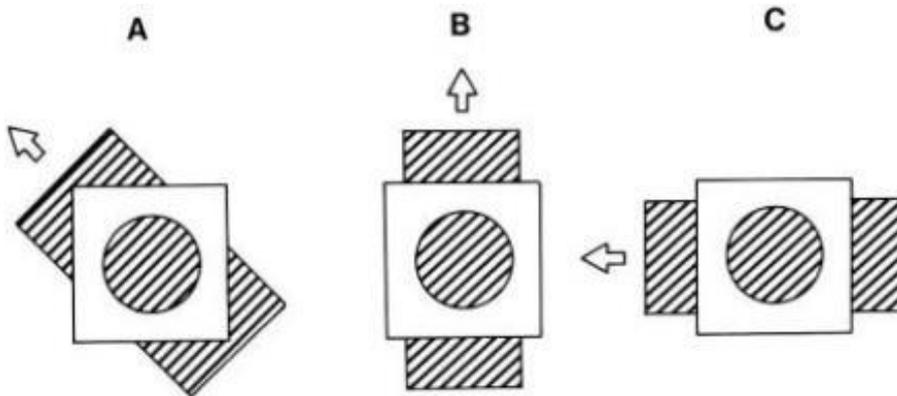


그림 3.18. 조리개 문제(출처: <https://stoomey.wordpress.com/2008/04/18/20/>)

외관의 불변성을 위반함으로써 발생할 수 있는 잠재적인 문제를 완화하는 한 가지 방법은 2차 불이익 함수[98]를 강건한 비용 함수[99, 100]로 교체하는 것이다.

3.5.3. 씬 플로우(scene flow)

자율 주행 차량에 있어 이미지 평면에서의 2차원 옵티컬 플로우보다는 객체의 실제 3차원 움직임이 더 필요하다는 점은 생각해 볼 가치가 있다. 2차원으로 표현된 것을 3차원으로 변환하는 것은 몇 가지 이유로 인해 그 중요성이 매우 크다.

- 장애물 탐지나 충돌 방지 같은 중요 임무를 보조하기 위한 3차원 정보를 얻기 위해



- 3차원에서 연속성에 대한 가정을 이용하기 위해
- 3차원 개체를 2차원 픽셀 위에 더 고차원적으로 표현하기 위해

KITTI의 2015 버전 씬 플로우(scene flow) 벤치마크에 따르면 씬 플로우 추정은 2개의 연속적인 스테레오 이미지 쌍(그림 3.19)을 기반으로 이뤄진다. 이때 이미지 쌍의 대응을 통해 지점의 3차원 위치뿐만 아니라 두 시간 사이의 3차원 움직임 또한 얻을 수 있다.

씬 플로우를 추정하는 데 있어 멘즈(Menze)와 가이저(Geiger)[101]는 슈퍼 픽셀을 활용해 움직이는 객체에 대한 3차원 매개변수 평면(그림 3.20)을 구축하는 방식을 제안했다.

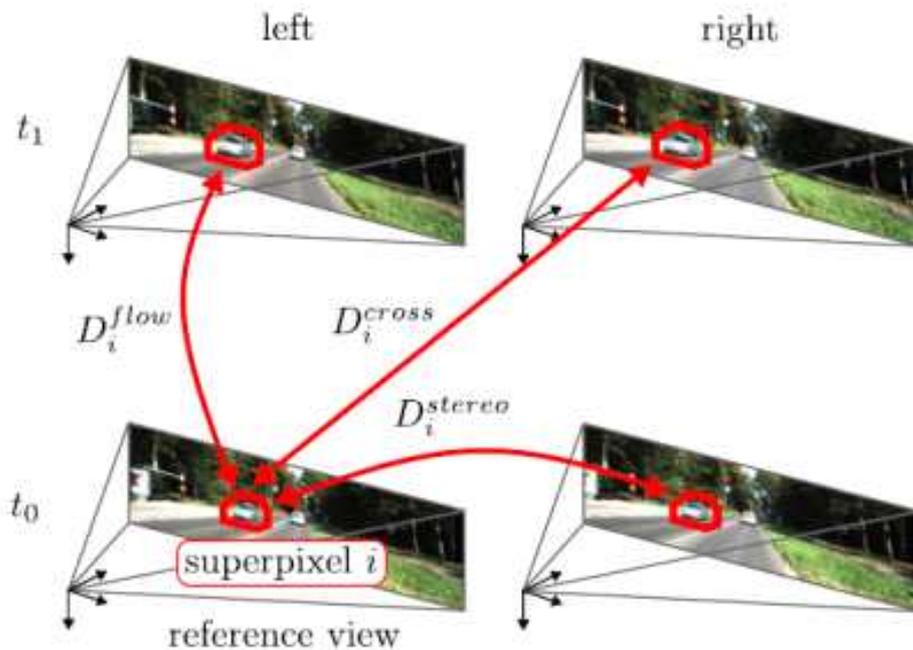


그림 3.19. 씬 플로우에 사용되는 2개의 스테레오 이미지 쌍[101]

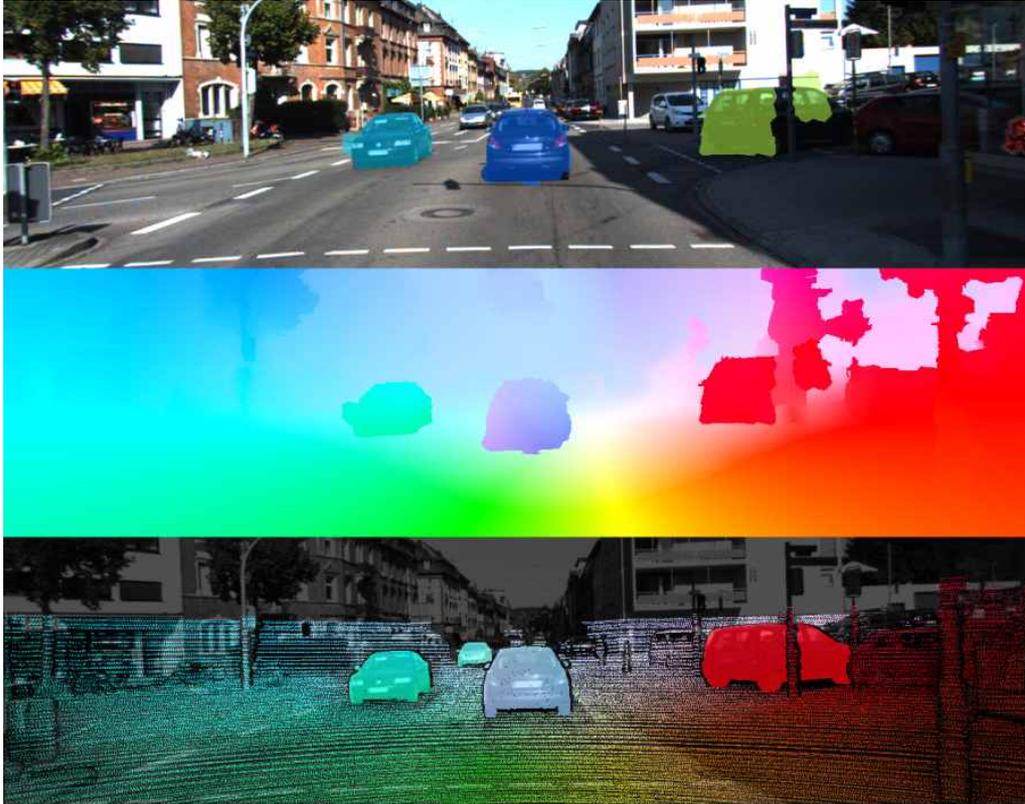


그림 3.20. 움직이는 객체의 추정, 옵티컬 플로우, 썬 플로우[101]

3.6. 객체 추적

객체 추적의 목적은 위치 속도 가속도 같은 객체의 상태를 시간에 대해 추정하는 것이다. 자율 주행 차량은 함께 교통 상황을 겪는 다른 객체를 추적해 안전 거리를 유지하고 각 객체의 궤적을 예측해야 한다. 보행자나 자전거를 타는 사람의 경우 특히 예측이 힘든데, 이는 보행자나 자전거의 방향이 갑자기 바뀔 수 있기 때문이다. 일반적으로 추적은 다음과 같은 이유로 인해 그 난이도가 올라간다.

- 객체는 종종 부분적으로(혹은 완전히) 가려져 있다.
- 같은 종류의 객체는 그 외관이 매우 비슷하다.
- 객체의 외관은 객체의 추적 중에도 자세 관절 조명 상황에 따라 매우 크게 바뀔 수 있다.

전통적으로 객체 추적은 순차 베이지안(sequential Bayesian) 필터 문제 형태로 구현된다.

1. 예측단계: 이전 시점에서 객체의 상태가 주어졌을 때 상태의 순간적인 변화를 정의하는 움직임 모델을 통해 현재 시간에 대한 객체의 상태를 예측한다.
2. 교정 단계: 현재 시간에 대한 객체의 상태가 예측됐고 센서의 측정치를 획득했

을 때, 현재 객체의 상태에 대한 사후 확률 분포를 관측 모델을 통해 계산한다. 관측 모델은 객체의 상태에 따라 어떻게 관측되는지를 나타낸다.

3. 1~2단계를 반복한다.

객체 추적에 있어 파티클 필터는 흔히 사용된다[102, 103]. 하지만 베이지안 필터는 재귀적인 특성을 갖고 있어 일시적으로 탐지에 실패했을 경우 정상 상태로 회복하기 힘들다. 재귀적이지 않은 방식으로 추적을 수행한다면, 추적을 글로벌 에너지 함수의 최소화로 볼 수 있다. 이때 움직임이 부드러워야 하고, 외관이 변하지 않음을 가정한다. 이러한 접근법은 객체의 가설과 한 객체가 따를 수 있는 궤적의 수가 크면 클수록 최적해를 구하기 위해 요구되는 계산량이 증가한다는 단점이 있다. 이를 해결하기 위해 사용되는 한 가지 방법은 바로 에너지 최소화에 대해 발견적인(heuristic) 방법을 사용하는 것이다[104].

객체 추적에 있어 흔히 사용되는 또 다른 방법은 탐지에 따른 추적(tracking-by-detection)이다. 연속적인 프레임에 대해 객체를 탐지하고 탐지된 객체는 프레임 전체에 걸쳐 연결된다. 두 단계 모두 어느 정도 불확실성을 보유하고 있다. 전자에 대해서는 탐지하지 못하거나 잘못된 위치를 탐지하는 경우가 있고, 후자에 대해서는 가능한 궤적 조합이 폭발적으로 증가하는 경우 데이터 결합 문제가 발생할 수 있다. 이러한 불확실성은 마르코프 결정 과정(MDP, markov decision process)를 통해 자연스럽게 처리할 수 있다. [105]에서 객체의 추적은 MDP를 통해 구현된다(그림 3.21).

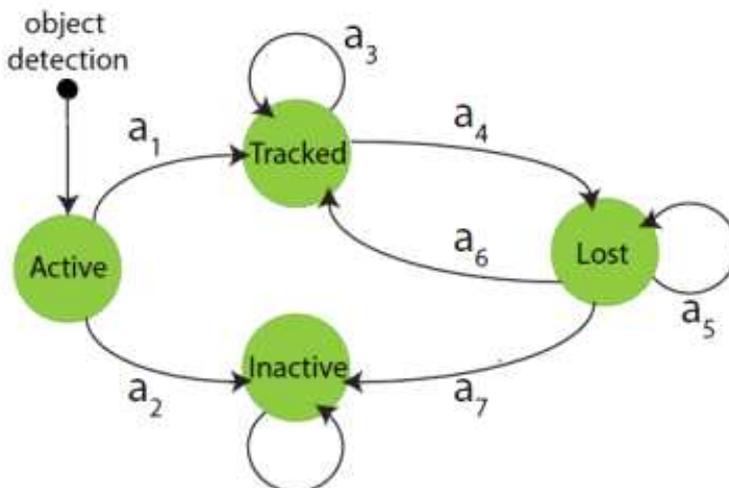


그림 3.21. MDP를 이용한 객체 추적의 구현(출처: [105])

- 객체는 총 네 종류 즉 활성화/비활성화/추적/유실 상태를 갖는다.

$$s \in S = S_{\text{활성화}} \cap S_{\text{추적}} \cap S_{\text{유실}} \cap S_{\text{비활성화}}$$



- 객체가 탐지됐을 때 ‘활성화’ 된다.
 - 탐지가 유효하다고 판단됐을 경우 해당 객체는 ‘추적’ 상태가 된다.
 - 탐지가 유효하지 않다고 판단됐을 경우 해당 객체는 ‘비활성화’ 상태가 된다.
 - ‘추적’ 상태의 객체는 ‘유실’ 될 수 있다.
 - ‘유실’ 된 객체는 ‘추적’ 상태로 환원될 수 있다.
 - 객체가 ‘유실’ 상태를 상당히 오래 유지할 경우 ‘비활성화’ 상태가 된다.
 - ‘비활성화’ 된 객체는 ‘비활성화’ 상태를 유지한다.
- 모든 움직임에 대해 $a \in A$ 를 만족한다.
 - 천이 함수 $T: S \times A \rightarrow A$ 는 입력과 출력이 정해져 있다.
 - 보상함수 $R: S \times A \rightarrow R$ 은 데이터를 통해 학습된다.
 - 정책 $\pi: S \rightarrow A$ 또한 데이터를 통해 학습된다.

이 알고리즘은 다음과 같이 동작한다.

- ‘활성화’ 상태에서는 탐지기가 제안한 객체의 후보들을 미리 훈련한 SVM을 통과시켜 그 유효성을 판단한다. SVM은 후보의 특징과 위치를 고려하고 a_1 과 a_2 중 어떤 행동을 할지 선택한다.
- ‘추적’ 상태에서는 실시간 외관 모델(online appearance model)을 통해 추적-학습-탐지 알고리즘을 기반으로 객체가 ‘추적’ 상태를 유지하거나 ‘유실’ 상태로의 전환을 결정한다. 이 외관 모델은 객체를 감싸는 가상의 상자를 템플릿으로 사용한다. 객체가 ‘추적’ 상태를 유지한다면 객체의 외관 모델을 지속적으로 갱신한다.
- ‘유실’ 상태에서는 객체가 ‘추적’ 상태에 있었을 때의 모든 템플릿을 수집한 다음 이를 통해 ‘추적’ 상태로 환원할지를 결정한다. 만약 객체가 제한된 시간 이상 ‘유실’ 상태를 유지할 경우 ‘비활성화’ 상태로 전환한다.

이 방법은 KITTI 데이터 세트에서 최고의 성능을 발휘했다. MDP의 예시는 그림 3.22에 묘사되어 있다.

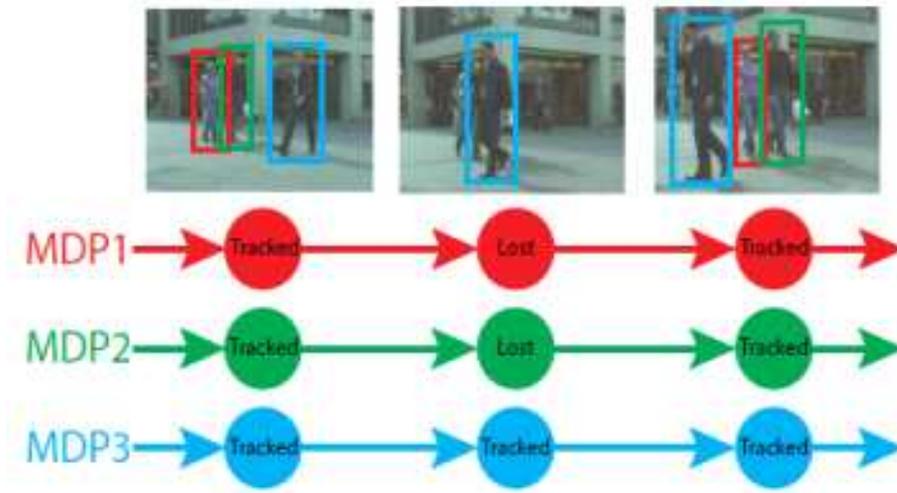


그림 3.22. MDP의 예제[105]

3.7. 결론

인지는 컴퓨터 비전에 크게 의존하는 자율 주행에 있어 핵심적인 요소 중 하나이다. 3장에서는 컴퓨터 비전 연구와 깊은 관련이 있는 분야에 대해 이야기했다. 문제의 정의, 데이터 세트, 일반적 접근법과 각각의 장단점을 다루어 적절한 이해를 유도했다. 딥 러닝을 통한 방법에 고무된 컴퓨터 비전 분야는 자율주행에 필요하다. 다양한 주제를 해결하는 데 있어 급격한 발전을 이루고 있다. 다양한 종류의 센서를 통한 주변 환경의 3차원 관측, 전체적인 신뢰도 및 정확성 또한 여전히 발전하고 있다. 여기서 핵심은 아무래도 각각의 시각적인 임무를 통합해 이따금씩 실패하는 인지를 절대 실패하지 않도록 만드는 것이다. 즉 원칙적으로 불확실성을 다룬다는 말이다. 이는 탐지, 분할, 인식, 추적, 운동 구조, 3차원 매핑, 위치 추정 등에 대해 더 나은 알고리즘을 꾸준히 연구할 것을 요구할 뿐만 아니라, 소프트웨어 및 하드웨어 모두에 대한 최적화와 구축, 시험 통합에 대한 공학적인 노력을 요구한다.

4. 예측 및 경로 계획

4.1. 개요

그림 4.1에서 볼 수 있듯이, 매핑 및 위치 추정(Map and Localization) 모듈은 포인트 클라우드(point cloud)나 GPS 같은 원본(raw) 센서 데이터를 받아서 이를 자율 주행차의 위치 정보로 변환해서 지도에 반영한다. 인지(perception) 모듈은 자율 주행차 주변에 있는 객체들을 감지한다. 이 두 모듈은 주변 세계를 객관적으로 인지하는 작업에 집중하는 반면, 경로 계획(routing), 트래픽 예측(traffic prediction), 동작 결정(behavioral decision), 모션 계획(motion planning), 피드백 제어(feedback control) 같은 모듈은 자율 주행차가 외부 환경의 변화를 예측해 주행 경로를 설정하는 과정을 주관적으로 처리한다는 점에서 구분된다.

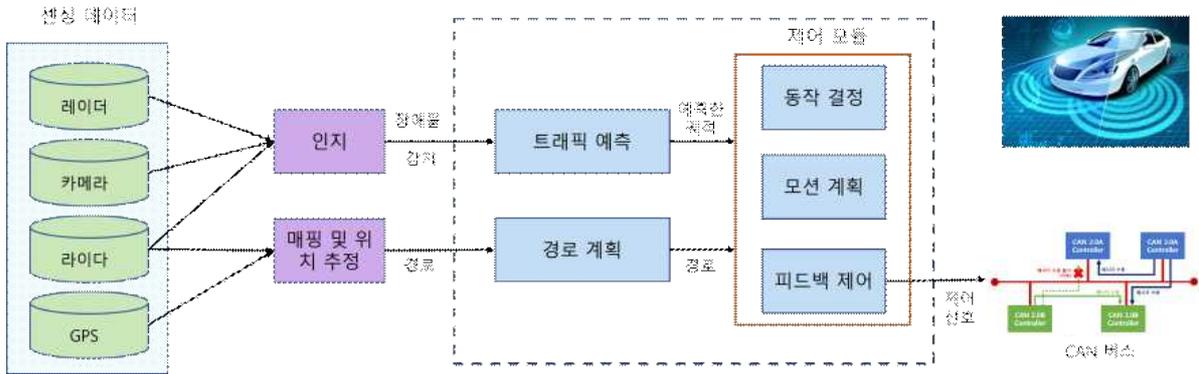


그림 4.1. 자율 주행차량의 시스템 구조

자율 주행 소프트웨어 시스템은 하드웨어와 소프트웨어로 구성된 복잡한 시스템이다. 자율 주행의 안전성과 신뢰성을 보장하려면, 컴퓨팅 하드웨어, 센서 융합, 인지 모듈, 트래픽 예측 모듈, 모션 계획 및 제어 모듈이 서로 잘 조율되어 작동해야 한다. 여기서 가장 중요한 것은 어떤 개별 모듈보다는 큰 문제를 작은 문제로 나눠서 각 문제에 적합한 모듈이 처리하도록 효과적으로 구성하는 것이다. 따라서 모듈사이의 협업, 그 중에서도 특히 인지 모듈과 계획 및 제어 모듈 사이의 협업이 대단히 중요하다.

그림 4.1에 나타낸 바와 같이 실시간 차량의 움직임을 생성하기 위해서는 경로 계획과 제어모듈은 실시간 동적 장애물 검출의 인식에 관한 입력 실시간 차량의 방위를 생성하는 위치추정에 관한 입력과 도로 기하구조와 정적 장애물들을 포착하는 지도 작성에 관한 입력을 결합하고, 이들 입력을 기반으로 한 차량의 동작 계획을 생성해야 한다.

그림 4.1에서 경로 계획 모듈은 도로 네트워크 정보에 대한 지도를 점검하고 경



로를 생성하는 모듈이다. 경로 계획은 단지 출발지부터 목적지까지 일련의 차로를 정확히 따라가는 것이라고 쉽게 생각할 수 있다. 물론 구글 맵스(Google Maps) 같은 지도 서비스에서 제공하는 내비게이션과 비슷한 점도 있지만, 자율 주행에서는 그보다 상세한 정보가 필요하며, 지도도 자율 주행에 최적화된 HD 지도를 주로 사용한다는 점에서 기존 내비게이션 서비스와는 근본적으로 차이가 있다.

일반적인 계획 및 제어 모듈의 가장 바깥쪽에는 트래픽 예측 모듈이 있다. 예측 모듈은 인지된 객체와 위치 속도(자동차, 자전거, 보행자 등과 같은) 유형을 비롯한 그 객체에 대한 ‘객관적인’ 속성을 함께 입력값으로 받는다. 그리고 입력된 인지 객체마다 예측 궤적을 계산해 그 결과를 다운스트림 동작 결정 모듈로 전달한다. 참고로 여기서는 예측 모듈에 입력한 값이 맵핑 및 위치 추정 모듈의 결과라고 명시적으로 표현하지 않았다. 그 이유는 맵핑 및 위치 추정 모듈은 매우 중요해서 거의 모든 모듈에서 자주 사용하기 때문에 어디에서나 편리하게 활용될 수 있도록 유틸리티 라이브러리로 구현하기 때문이다.

예측 궤적은 공간 정보와 시간 정보를 모두 담고 있으며 다운스트림 모듈에서 출력 값을 계산하는 데 꼭 필요한 기초 정보다. 이전 연구 [107]에서는 예측 모듈이 보조 소프트웨어 라이브러리로 구현했는데 인지 모듈에서 사용해 출력값을 세밀하게 보정 하거나, 결정 및 계획 모듈에서 감지된 입력 객체를 전처리하기 위한 용도로 활용했다. 예측 모듈을 소프트웨어 라이브러리로 구현하면 주기적으로 업스트림 모듈로부터 데이터를 가져와서 다운스트림 모듈로 데이터를 발행할 필요가 없고, 상태나 메모리를 따로 관리할 필요가 없다는 장점이 있다. 소프트웨어 라이브러리로 만들면 필요한 함수를 호출해서 계산 결과를 받도록 일부 API만 공개해도 된다. 이러한 이유로 계산 논리의 측면에서 소프트웨어 라이브러리 형태가 대체로 덜 복잡하다. 반면 자율 주행 시스템을 구성하는 모듈로 만들면 업스트림 모듈의 데이터를 받아서 계산한 결과를 프레임 단위로 주기마다 다운스트림 모듈로 보내야 하는 부담이 있다. 모듈은 대체로 계산 정확도를 높이기 위해 이전 프레임의 상태 정보를 유지해야 한다.

자율 주행 기술의 발전으로 실제로 자율 주행차가 도로를 주행하는 것이 현실이 되면서, 이를 구현하는 업계에서 트래픽 예측의 중요성에 주목하기 시작했다. 따라서 최신 시스템에서 사용하는 트래픽 예측 모듈은 소프트웨어 라이브러리 형태보다는 내부 모듈 형태로 구현하는 것이 대부분이다[108-111]. 이 장에서는 트래픽 예측 문제를 머신러닝 기반으로 접근해서 푸는 최신 기법을 소개한다.

트래픽 예측 모듈에 바로 연결되는 다운스트림 모듈은 자율 주행 시스템에서 ‘부조종사(co-pilot)’의 역할을 하는 동작 결정 모듈이다. 이 모듈은 트래픽 예측 모듈과 경로 계획 모듈로부터 받은 값을 모두 입력 값으로 사용한다. 동작 결정 모듈은 이러한 입력값을 토대로 자동차를 조작하는 명령을 생성한다. 이러한 명령



의 예로, ‘현재 차선에서 전방 자동차를 따라간다’, ‘신호등 앞의 정지선에 멈춰서 보행자가 지나갈 때까지 기다린다’, ‘정지 표지판을 보면 다른 자동차에게 양보한다’ 등이 있다.

동작 결정은 자율 주행차 자체의 동작 뿐만 아니라, 인지하거나 지도에 나온 객체에 대한 동작도 결정한다. 좀 더 구체적인 예를 들면 자율 주행차가 현재 달리는 차로에서 어떤 객체를 감지했는데, 경로 계획 모듈은 자율 주행차에게 현재 차로를 유지하라고 알려주는 경우를 생각해보자. 자율 주행차 자체에 대해서는 차로를 유지하라고 결정(합성 결정(synthetic decision))할 수 있지만 전방에 감지된 자동차에 대해서는 그 차를 뒤따라가라고 결정(독자 결정(individual decision))할 수 있다. 이렇게 인지된 장애물마다 결정된 동작은 모션 계획 단계에서 적용할 최적화 제약사항과 비용으로 변환된다.

자율 주행차 자체에 대한 동작은 개별적인 동작 결정을 모두 취합해서 결정하기 때문에 합성 결정이라 부른다. 합성 결정은 주로 모션 계획 단계에서 최종 상태의 동작 조건을 결정하는 데 필요하다. 동작 결정 모듈에서 출력한 명령 집합의 구체적인 형태는 구현 방식에 따라 달라질 수 있다. 최신 자율 주행 자동차 시스템에서는 동작 결정 모듈을 개별 모듈로 설계 및 구현하는 추세다. 그러나 모션 계획 모듈 같은 다운스트림 모듈에 동작 결정 모듈의 논리와 역할을 통합하는 방법도 있다[13, 64, 107].

앞에서 설명했듯이 모듈 사이의 협업이 중요하기 때문에 업스트림 동작 결정 모듈과 다운스트림 모션 계획 모듈의 로직은 일관성을 유지해야 한다. 다시 말해, 모션 계획 모듈은 동작 결정 모듈의 출력값을 정확히 따라서 자율 주행 자동차에 대한 궤적을 계획할 때 이를 수행 해야 한다. 동작 결정 모듈의 명령 집합은 최대한 다양한 트래픽 동작 시나리오를 다뤄야 하는데 그 내용이 반드시 완벽해야 한다는 의미는 아니다. 사람조차도 정확히 판단하지 못하고 막연한 감으로 충돌 회피 동작을 수행하는 모호한 상황이 발생하기 마련이다. 동작 결정 모듈의 명령 집합을 명확하게 표현하면 오류 진단 및 디버깅할 때는 좋지만 정작 중요한 부분은 이러한 동작 결정을 모션 계획 단계에 적용할 구체적인 제약 조건이나 비용으로 변환하는 데 있다. 합리적인 독자 결정을 할 수 없는 최악의 시나리오에서는, 충돌 회피를 위한 암묵적 비용이 모션 계획을 위한 기본적인(default) 독자 결정이 된다.

모션 계획 모듈의 역할을 간단히 표현하면, 계획한 경로나 궤적을 따라 A 지점에서 B 지점으로 이동하는 방법을 찾는 것이다. 만약 현재 위치가 A 지점이고, B 지점이 목표 지점(예: 차로 시퀀스상의 어떤 목표 지점)이라면 모션 계획은 A 지점에서 B지점으로 가는 로컬 경로를 탐색하는 최적화 문제라고 볼 수 있다. 모션 계획 모듈은 동작 결정 모듈의 출력 값을 제약 조건(constraints)으로, 경로 계획



모듈의 출력 값을 목표(goal)로 받는다.

모션 계획 모듈은 동작 결정 모듈보다 구체적인 문제를 결정한다. 모션 계획 모듈은 위치, 방향, 속도, 가속도, 곡률뿐만 아니라 이러한 속성의 고차 미분 값으로 구성된 궤적 지점을 반드시 지나도록 궤적을 계산해야 한다. 앞서 설명했듯이 모듈 사이의 협업이 중요하기 때문에 모션 계획 모듈은 다음과 같은 두 가지 규칙을 반드시 따라야 한다. 첫째, 연속된 계획 사이클 사이에서 계획한 궤적의 일관성이 유지되어야 한다. 즉, 외부 요인이 크게 변하지 않는다면 연속된 두 사이클의 궤적도 급격히 변하지 않아야 한다. 둘째 계획한 궤적을 다운스트림 피드백 제어 모듈에서 수행할 수 있도록 보장 해야 한다. 이는 주로 곡률이나 곡률의 미분 값 같은 속성이 물리적 제어 한계를 벗어나지 않도록 연속적이며 매끄러워야만 한다는 뜻이다.

그림 4.1을 보면 인지 모듈의 출력값 뿐만 아니라 맵핑 및 위치 추정 모듈의 출력 값도 모션 계획 모듈과 동작 결정 모듈에 직접 입력되는 것을 알 수 있다. 얼핏 보면 중복된 설계처럼 생각할 수 있다. 하지만 트래픽 예측에 대한 일종의 백업 역할을 해서 보안성을 높이는 데 도움이 된다. 또한, 트래픽 예측 작업을 수행하는 도중에 새로운 장애물들이 감지될 수 있다. 예측에 실패할 때뿐만 아니라 예측하는 동안 새로운 장애물이 나타날 때도 쉽게 접근할 수 있는 맵핑 및 위치 추정 유틸리티 라이브러리와 함께 이렇게 중복된 인지 정보를 갖고 있으면, 동작 결정 모듈과 모션 계획 모듈이 충돌 회피 동작을 결정하는 데 필요한 최소한의 객체 정보를 가질 수 있다.

그림에서 가장 아래쪽에는 피드백 제어 모듈이 있다. 이 모듈은 CAN 버스를 통해 자동차를 직접 제어한다. 제어 모듈의 핵심 역할은 계획된 궤적에 담긴 궤적 지점 정보를 가져와서, 브레이크 스티어링휠 스톱을 조작하기 위한 드라이브 바이 와이어(drive-by-wire) 신호를 계산하는 것이다. 이러한 계산 작업은 주로 실제 자동차가 계획된 궤적에 최대한 가깝게 주행하려는 목적으로 수행하며 이때 자동차 및 도로의 물리적 모델도 함께 고려 한다.

지금까지 설명한 모듈은 자율 주행차 계획 및 제어에서 가장 핵심적인 모듈이다. 이렇게 구분하는 이유는 자율 주행차의 계획 및 제어에 관련된 복잡한 문제를 여러 개의 작은 문제로 효과적으로 나누는 데 있다[13, 107]. 각 모듈마다 자신이 맡은 문제에 집중할 수 있게 만들면 모듈화 및 병렬화를 통해 자율 주행 자동차 소프트웨어 개발의 복잡도를 크게 줄일 수 있고 연구 개발의 효율성도 크게 높일 수 있다. 지금까지 설명한 방법의 장점이 바로 여기에 있다. 본질적으로, 동작 결정 모듈, 모션 계획 모듈, 피드백 제어 모듈은 같은 문제를 서로 다른 수준에서 해결하는 셈이다. 그리고 데이터 흐름에 따라 위치한 각 모듈 사이의 계산 결과는 서로 밀접하게 연결돼 있다. 이러한 모듈을 구현할 때 중요하게 고려할 사항은 계



산을 일관성있고 논리정연하게 구성하는 것이다.

4.2. 트래픽 예측

트래픽 예측 모듈(또는 간단히 예측 모듈)은 계획 및 제어 모듈과 직접 연결되는 모듈로서, 현재 감지된 객체의 가까운 미래의 동작을 예측하는 역할을 담당하며, 예측 결과와 함께 시공간 궤적 지점들에 대한 세부 정보를 만들어서 이를 제어 모듈에 전달한다.

인지 단계에서 감지된 장애물은 대부분 위치, 속도, 방향, 가속도 등의 속성들을 갖고 있다. 이러한 속성은 운동학적(kinematic) 특성에 가까워서 간단히 물리 규칙을 적용하면 적절한 수준의 예측값을 즉시 구할 수 있다. 하지만 트래픽 예측의 목적은 단순히 주어진 물리적 속성으로부터 즉각적인 예측값을 구하는 것이 아니라, 몇 초 주기에 발생하는 동작을 예측하는 것이다. 이러한 예측값을 구하려면 동작 히스토리, 주변 시나리오, 지도 특징 같은 다양한 요소를 고려해야 한다. 예를 들어 그림 4.2처럼 교차로에 있을 때 자동차가 계속 직진할지, 아니면 우회전할지, 또는 인도에 있던 보행자가 횡단보도를 건널지 아니면 기다리고 있을지와 같은 동작을 트래픽 예측 모듈로 예측해야 한다. 이러한 동작 예측 작업은 주로 범주형 (categorical) 문제에 해당하므로 이를 분류(classification) 문제로 구성해 머신 러닝 기법을 적용하는 방식으로 해결할 수 있다[112-114]. 하지만 단순히 동작 관점에서 예측하는 것만으로는 부족하다. 앞에서 설명했듯이 트래픽 예측 모듈의 실제 출력 값은 시간 정보, 속도, 방향 등의 속성들을 가진 궤적 지점으로 구성된 예측된 궤적이기 때문이다. 따라서 트래픽 예측 문제를 다음과 같은 두 가지 문제로 나눠서 처리한다.

- 범주형 도로 객체 동작에 대한 분류 문제: 예를 들어 자동차가 차로를 변경하거나 현재 차로를 유지할지 여부, 또는 보행자가 횡단보도를 건널 것인지 여부 등
- 속도 및 시간 정보를 통해 예측 경로를 생성하는 회귀 문제: 예를 들어 교차로를 지나갈 때 보행자의 속도 변화는 크지 않지만, 자동차가 방향을 틀 때는 감속했다가 가속해야 하는데 구체적인 속도 변화는 회전 반경의 길이와 곡률에 따라 결정 된다.



그림 4.2. 도로에 있는 객체에 대한 트래픽 예측(출처: [115])

4.2.1. 분류를 이용한 동작 예측

도로에 있는 객체의 동작을 예측하는 문제는 주로 객체의 유형에 따라 달라진다. 자동차의 동작은 주로 ‘현재 차로 유지’, ‘방향 전환’, ‘차로 변경’ 등 몇 가지로 구분할 수 있는 반면, 자전거나 보행자의 동작은 매우 다양하다. 이러한 동작의 차이 때문에, 머신 러닝 기반 모델에는 보통 단일 모델을 적용하지 않고 각 유형에 적합한 맞춤형 모델을 사용한다. 자동차의 동작이 보행자의 동작보다 더 구조적이기 때문에, 본 고에서는 자동차의 동작 예측을 중심으로 소개한다.

사실 자동차에 대한 동작 예측 문제도 쉽지는 않다. 처음에는 분류 모델을 차로 유지, 차로 변경, 방향 전환 같은 동작의 유형마다 따로 만들어야 한다고 생각하기 쉽다. 하지만 이렇게 할 경우 확장성이 떨어진다. 실제 지도는 상당히 복잡하기 때문이다. 예를 들어 우회전이나 좌회전 차로가 여러 번 나올 수 있고, 교차로가 항상 네 방향인 것은 아니다. 게다가 차로가 서서히 오른쪽으로 꺾일 수도 있고, 교통 규칙을 위반하지 않고서는 선택할 동작이 없을 수도 있다. 이처럼 지도나 시나리오에 따라 동작의 유형을 선택하게 하면 분류 유형 (분류 문제의 표현으로 ‘레이블’)이 엄청나게 복잡하고 확장성이 떨어지기 때문에 이런 식으로 동작의 유형을 선택할 수 없다.

도로를 주행하는 자동차의 동작을 예측하는 문제는 ‘자동차가 주행할 차로를 선택하는’ 이진 분류 문제로 구성할 수 있다. 이렇게 새로운 방식으로 추상화하



면, (우회전, 좌회전, 차로 변경 등과 같은) 특정한 차로 시퀀스를 선택할 때 문맥이 어떻게 변하는지에 대해 더는 신경 쓰지 않아도 된다. 또한 머신러닝 모델을 (이진 분류 모델이라는) 하나의 통합된 모델로 만들 수 있다. 이 모델은 차로 정의와 분할에 대한 단 하나의 종속성이 발생하는데 자동차는 대부분 차로를 따라 주행해서 예측 궤적을 벗어날 수 없기 때문에 이 종속성은 불가피하다.

일반적으로 자동차가 차로의 특정 논리적 또는 구조적 시퀀스를 따른다고 가정한다. 얼핏 생각하면 이 가정은 과도한 제약을 가한다고 생각할 수 있다. 하지만 머신러닝을 적용할 때는 아무런 제약이 없는 모든 동작을 학습시키는 것보다, 나름 합리적으로 구조화된 동작부터 학습을 시작하는 것이 굉장히 중요하다. 또한 머신러닝 기법을 적용할 때는 방대한 양의 데이터가 쌓이는데 이로 인해 선택할 수 있는 학습 모델의 복잡도에 제약이 발생한다. 따라서 자율 주행에서 트래픽 예측을 수행할 때, 이러한 로직과 합법적인 동작부터 먼저 파악하는 것이 좋다. 실제로 논리적으로 연결되지 않은 차로들을 추종하는 ‘잘못된’ 동작이 발생할 수도 있다. 그러나 일단 정상적인 동작에 대해 잘 만든 모델을 구축하고 학습 데이터도 충분히 갖추고 있다면, 학습 모델에 이러한 비정상적인 동작이 포함되지 않도록 쉽게 조절할 수 있다. 결론적으로 차로 시퀀스 기반으로 구축한 트래픽 예측 모델은 다양한 경우에 상당히 효과적인 기법이라고 볼 수 있다.

4.2.1.1. 자동차 동작 예측을 위한 특징 설계

앞에서 설명했듯이 머신러닝 기반 분류기(classifier)가 잘 작동하려면 차로 시퀀스 기반의 분류 문제에 대한 레이블링과 특징 설계가 대단히 중요하다. 자동차 동작 예측 문제를 다룰 때는, 자동차 히스토리 특징, 차로 시퀀스 특징, 주변 객체 특징이라는 세 가지 유형의 특징을 고려한다. 그럼 각 유형을 하나씩 살펴보자.

- 자동차 히스토리 특징은 자동차가 현재 차로나 이전 차로를 그동안 어떻게 이동했는지를 히스토리 형태로 나타낸다.
- 차로 시퀀스 특징: 확장 차로 시퀀스(extended lane sequece)란, 자동차가 이용할지 여부를 분류할 대상 중 한 인스턴스다. 그러므로 여기서 지정된 차로 시퀀스에서 일부 지점들을 샘플링한다. 이렇게 샘플로 추출한 ‘차로 지점(lane point)’ 마다 확장 차로 시퀀스의 모양을 나타내는 속성을 계산한다. 예를 들어 방향(heading), 곡률(curvature) 시퀀스 특징으로 추출된다. 이러한 특징 유형은 선택할 차선 시퀀스의 모양을 표현 한다.
- 주변 객체 특징: 이 특징 유형은 예측에 사용할 자동차 주변의 객체를 표현하며, 다른 특징보다 계산하기 힘들다. 그 이유는 차로 모양이나 자동차의 자세 히스토리 뿐만 아니라 주변 객체도 자동차의 향후 동작을 결정하기 때문이다. 예를 들어, 현재 주행하는 차로의 왼쪽과 오른쪽 차로에 예측할 자동차를 투영한다. 그



러면 전방 거리를 계산할 수 있다.

개념의 이해를 돕기 위해 간단히 예를 들면, 어떤 자동차가 서서히 오른쪽을 향하면서 차로의 경계 가까이 다가온다고 하자. 그러면 자동차의 진행 방향을 바꿀 정도로 눈에 띄는 주변 객체가 없다면, 직진을 표현하는 차로 시퀀스보다 우회전을 표현하는 차로 시퀀스를 선택해야 할 가능성이 훨씬 높다.

4.2.1.2. 자동차 동작 예측을 위한 모델 선택

앞에서 언급한 특징들이 자동차 및 그 주변에 관련된 정보를 대부분 포함하지만, 완벽한 특징 집합을 제시한다기보다는 내 경험에 비춰볼 때 현실적이라고 생각하는 특징 집합을 제시하려는 의도로 설명한 것이다. 또한 머신러닝에서 특징은 모델과 밀접한 관련이 있다. 자동차의 동작 예측에 적용할 수 있는 모델은 다음과 같이 두 가지가 있다.

- 비기억 모델(memory-less model): SVM(Support Vector Machine)[116], DNN(Deep Neural Networks)[117] 등이 여기에 해당한다. 이러한 모델은 한 번 학습되면 그대로 유지되기 때문에 비기억 모델에 해당한다. 또한 출력 값은 이전 입력 인스턴스에 영향을 받지 않는다. 비기억 모델로 히스토리 정보를 수집하려면 그 정보를 특징으로 명확하게 인코딩해야 한다. 예를 들어 자동차 정보에 대한 여러 히스토리 프레임을 받아서, 각 프레임에서 특징을 추출하고, 그 결과를 예측뿐만 아니라 학습에도 활용할 수 있다.

- 기억 모델(memory model): RNN(Recursive Neural Networks) 구조[118, 119]를 가진 LSTM(Long Short-Term Memory) 모델이 여기에 해당한다. 이 모델은 출력값이 입력값의 영향을 받기 때문에 메모리를 갖는다. 하지만 기억 모델은 학습시키기 훨씬 어렵다 RNN 모델에 주어진 입력값이 단지 현재 자동차의 특징 및 현재 주변 객체의 특정처럼 현재 프레임 정보만 표현되어 있더라도, RNN 모델은 어떻게든 모델 매개변수로 이전 입력값들을 ‘기억’ 해서, 이전 입력값이 현재 출력값에 영향을 미치게 된다.

어떤 모델을 선택할지는 구체적인 시나리오마다 다르다. 맵핑과 주변 환경이 그리 복잡하지 않은 시나리오에서는 간단히 비기억 모델을 사용해도 충분하다. 하지만 교통 상황이 매우 복잡해서 히스토리 정보를 완벽하게 다루기 위해서는 RNN 같은 기억 모델을 활용해야 한다. 학습 및 튜닝이 훨씬 더 어렵다면, 온라인 시스템으로 구현하는 것이 더 편하다. 기억 모델은 입력 값으로 단지 현재 정보만 이용하고 히스토리 정보는 기억하기 때문이다. 반면 비기억 모델은 온라인 시스템으로 구현하기에는 좀 더 복잡하다. 이렇게 온라인 시스템으로 만들면 특징 추출에



사용되는 히스토리 데이터도 온라인으로 구성해야 한다. 이론적으로 타임 윈도우 바' 는 인지 모듈이 객체를 추적하는 데 허용된 최대 시간이다. 트래픽 예측 모듈은 대부분 이러한 w값을 5초와 같은 고정된 길이로 설정한다. 예측 궤적은 반드시 최소 거리나 최소 시간 중 하나를 기준으로 삼아야 한다. 만약 '메모리' 길이로 5초를 선택하면, 예측 궤적은 최대 5초를 기준으로 삼게 되지만, 3초와 같이 더 짧은 타임 윈도우를 선택했을 때보다 신뢰성이 더 높아진다. 향후 동작을 정확히 예측하기 위해 얼마나 많은 히스토리 데이터가 필요한지에 대한 정확한 이론적인 한계는 본 고에서 다루지 않는다. 동작 예측에서 중요한 측정 기준은 정밀도와 재현율이다. 정밀도(precision)는 예측 궤적 중에서 실제로 자동차가 선택한 궤적의 수를 의미한다. 그리고 재현율(recall)은 실제 동작 궤적 중에서 예측한 궤적의 수를 나타낸다. 트래픽 예측 모듈은 각 프레임마다 예측한 궤적을 출력하기 때문에, 이러한 두 가지 측정 기준은 모든 프레임에 대해 예측된 궤적을 취합해서 계산한다.

4.2.2. 자동차 궤적 생성

자동차의 동작이 결정되면 예측 모듈은 예측 차로 시퀀스를 추종하는 실제 시공간 궤적을 생성해야 한다. 간단한 방법은 물리적 규칙과 특정한 가정을 이용하는 것이다. 여기서는 자동차의 차로 기반 지도 좌표를 추적하기 위해 칼만 필터를 소개 한다.

이 방법의 핵심 가정은 자동차가 차로의 중심선(기준선 reference line)을 따라가는 것이다. 따라서 궤적상에서 자동차의 예측 지점들에 대한 (s, l) 좌표들을 추적하는 데 칼만 필터를 사용한다. 간단히 말해 s는 차로의 기준선에 대한 거리(종방향 거리)를 나타내고, l은 어떤 지점에서 s방향에 수직인 거리 (횡방향 거리)를 나타낸다. 예측 차로 시퀀스마다 칼만 필터로 주어진 예측 궤적을 추적한다.

규칙 기반 방법 뿐만 아니라 머신러닝 기반 방법으로도 궤적을 생성할 수 있다. 궤적 생성에 머신러닝 기반 모델을 적용하면 실제 궤적 히스토리를 활용하기 때문에, 규칙 기반으로 생성한 궤적보다 실제 히스토리에 좀 더 가까운 궤적을 생성할 수 있다. 이때 회귀 모델(regressions model)이 적합하다. 자동차 히스토리 정보를 특징으로 입력해 자동차의 실제 경로를 도출하기 위한 모델을 구축할 수 있다. 그러나 실제 궤적은 모델링하기 훨씬 더 어려우며 동작 자체보다는 덜 중요하다.

이 절에서는 트래픽 예측 문제를 두 단계로 구성했다. 먼저 동작을 예측한 다음, 실제 궤적을 계산한다. 첫 번째 단계인 동작 예측은 모든 차로 시퀀스에 대한 이진 분류 문제로 명확히 표현할 수 있는 반면, 실제 시공간 궤적을 계산하는 두 번째 단계는 모션 계획의 특정 기법을 사용해야 한다. 동작 예측 문제에서 다양한



자동차들 간의 상호 영향을 도입하면 복잡도가 폭발적으로 증가하기 때문에, 지금까지는 각기 다른 대상 자동차들 간의 상호작용을 명확하게 고려하지 않았다. 하지만 예측 빈도가 충분히 높다면 대상 자동차들 간의 상호작용이 어느 정도 반영된다.

4.3. 경로 계획(routing)

경로 계획은 구글 맵스 같은 내비게이션 서비스와는 상당한 차이가 있다. 기존의 내비게이션 서비스는 지점 A로부터 지점 B까지 가는 도로 구간을 찾는 방식으로 실행된다. 이러한 내비게이션 서비스의 최소 요소는 도로의 차로다. 도로와 차로는 도로 표지판이나 경계에 따라 쉽게 정의할 수 있다. 자율 주행차의 경로 계획 문제도 지점 A로부터 지점 B까지 가는 경로를 찾는 것은 마찬가지지만 출력값은 사람이 아닌 동작 결정 및 모션 계획 모듈의 입력으로 전달하기 적합하게 구성해야 한다. 따라서 차로 수준 경로 계획은 HD 지도에 정의된 차로 수준을 따라야 한다 HD 지도에 정의된 차로는 실제에 맞게 분할된 차로나 도로와 다르다.

효과적인 경로 계획 알고리즘은 네 가지 기준을 충족해야 한다. 첫째, 모션 계획 기법은 현실적인 정적 환경에서 항상 최적의 경로를 찾을 수 있어야 한다. 둘째, 동적 환경으로 확장 가능해야 한다. 셋째, 선택된 자기 참조 접근법과 호환성을 유지하고 강화해야 한다. 넷째, 복잡성, 데이터 스토리지 및 계산 시간을 최소화해야 한다[120]. 이 절에서는 자율 주행차에 적용할 수 있는 가장 자주 사용되는 경로 계획 알고리즘의 개요를 제시하고 정적/동적 환경에 가장 적합한 알고리즘에 대해 논의한다.

4.3.1. 가중치 방향성 그래프

자율 주행용 경로 계획의 두드러진 특징은 자율 주행 자동차가 하기 힘든 동작도 함께 고려해서 경로를 결정해야 한다는 것이다. 바로 이 점이 구글 맵스 같은 내비게이션 서비스와 구분되는 가장 큰 차이이다. 예를 들어 자율 주행용 경로를 설정할 때 모션 계획 모듈에서 차로 변경을 하려면 더 많은 공간과 시간이 필요하다고 알릴 경우 차로 변경을 하지 않는다. 그 이유는 안전하지 않은 단거리 차로 변경에 대한 경로 계획 세그먼트를 생성하지 않아야 하기 때문이다. 따라서 이렇게 위험한 경로에 대해서는 높은 ‘비용’을 할당해야 한다. 정리하면 자율 주행 자동차가 어렵게 느끼는 동작은 사람이 운전할 때와 상당히 다르기 때문에 경로 계획 모듈은 자율 주행차의 모션 계획 모듈에 맞춰 정의해야 한다. 이러한 의미에서 자율 주행차에 대한 경로 계획 출력값은 사람이 쓰는 내비게이션의 출력값과 얼마든지 다를 수 있다.



자율 주행차에 대한 HD 지도 기반 경로 계획 문제는 가중치 방향성 그래프에서의 최단 경로 탐색 문제로 추상화할 수 있다. 형식적인 용어에서 방향성 그래프는 하나의 순서쌍 $G=(V, E)$ 이다. 이 때 G 는 정점이라 부르는 원소를 갖는 집합이고 E 는 방향성 에지라 하는 정점들의 순서쌍 집합이다. 이는 통상적으로 에지라 부르는 정점들의 비순서쌍으로 정의되는 단일 방향성 그래프와는 다른 것이다. 가중치를 갖는 방향성 그래프의 예를 그림 4.3에 도시한다.

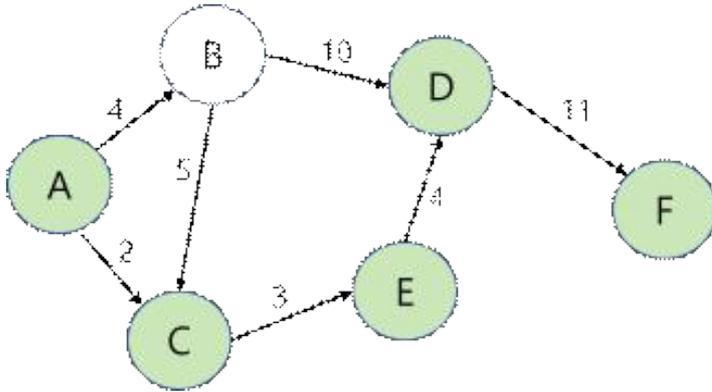


그림 4.3. 가중치를 갖는 그래프 데이터 구조

가중치를 갖는 방향성 그래프는 도로 네트워크를 나타내는데 사용될 수 있다. 가령 하나의 정점은 샌프란시스코를 나타낼 수 있고 또 다른 정점은 NewYork을 나타낼 수 있다고 하면, 이들 두 정점을 연결하는 에지는 두 도시 간 거리의 기록에 해당한다. 라우터 알고리즘은 두 도시 간의 최단 경로를 탐색하기 위해 도로 네트워크 그래프에 적용될 수 있다.

4.3.2. 다익스트라(Dijkstra) 알고리즘

다익스트라 알고리즘은 소스에 가장 가까운 정점 중 소스에서 정점까지의 최단 경로를 계산하는 하위 문제를 해결함으로써 작동한다[121]. priority-min 큐에서 새로운 정점을 유지함으로써 다음으로 가까운 정점을 찾고 오직 하나의 최단 경로만 찾을 수 있도록 하나의 중간 노드만 저장한다.

전통적인 다익스트라 알고리즘은 경로 계획을 위한 탐욕스러운 전략에 의존한다. 이 알고리즘은 솔루션이 적용되는 애플리케이션에 관계없이 그래프에서 가장 짧은 경로를 찾는 데 사용된다. 수정된 다익스트라 알고리즘은 그럴듯한 최단 경로를 생성하는 데 드는 비용이 큰 상황에서 대체 경로를 찾는 것을 목표로 한다. 이 수정된 알고리즘은 그래프의 각 가장자리를 따라 자유 상태를 정의하는 확률의 형태로 고전 알고리즘에 또 다른 구성요소를 도입한다[122]. 이 기술은 참조 알고리즘의 계산적 단점을 극복하는 데 도움이 되며, 새로운 응용 프로그램에서



사용할 수 있도록 지원한다.

또 다른 개선된 다익스트라 알고리즘은 소스 노드로부터 동일한 거리의 모든 노드를 중간 노드로 예약한 다음 목표 노드로 성공적으로 통과할 때까지 모든 중간 노드에서 다시 검색한다. 반복을 통해 가능한 모든 최단 경로가 발견되고 평가될 수 있다[123].

다익스트라 알고리즘은 이전에 건너간 노드로부터 떨어진 곳에 데이터를 저장할 수 없다. 이러한 단점을 극복하기 위해 데이터의 저장을 위한 다층 사전(multi-layer dictionary) 알고리즘[124]이 도입되는데, 이는 두 개의 사전과 계층적 순서로 구성된 데이터 구조 목록으로 구성된다. 첫 번째 사전은 각 노드를 인접 노드에 매핑한다. 두 번째 사전은 각 인접 경로의 경로 정보를 저장한다[124].

다층 사전은 글로벌 항법 위성 시스템(GNSS) 좌표와 나침반 방향을 신뢰할 수 없는 실내 환경 애플리케이션에서 다익스트라 알고리즘을 위한 포괄적인 데이터 구조를 제공한다. 데이터 구조의 경로 정보는 로봇이 각 노드 또는 접합에서 실행해야 하는 회전각의 정도를 결정하는 데 도움이 된다. 제안된 알고리즘은 길이 측면에서 최단 경로를 생성하는 동시에 필요한 최소 총 회전 각도(도) 측면에서 가장 탐색 가능한 경로를 제공하는데, 이는 전통적인 다익스트라 알고리즘 내에서 계산하기가 불가능하다[124].

플로이드 알고리즘은 양 또는 음의 가중 그래프에서 최단 경로를 찾는 데 널리 사용되는 그래프 알고리즘인 반면, 다익스트라는 양 가중 그래프에서 단일 소스(소스 정점에서 다른 모든 정점까지 최단 경로 찾기) 최단 경로를 찾는 데 가장 잘 작동한다[125].

다익스트라는 경로 계획을 위한 신뢰할 수 있는 알고리즘이다. 또한 최단 경로를 결정하기 위해 가능한 모든 결과를 계산해야 하고 음의 에지를 처리할 수 없기 때문에 사용하는 메모리 양이 많다[126]. 그것의 계산 복잡도는 $O(n^2)$ 이며, n 은 네트워크의 노드 수이다[127]. 그 한계로 인해 응용 프로그램을 기반으로 많은 개선된 변형들이 생겨났다. 앞서 메모리 단점에 대해 논의했듯이, 새로운 메모리 체계가 도입되었다[124]. 막대한 비용 요소를 사용하여 매핑할 수 있는 솔루션도 있었다[122].

우리는 다익스트라 알고리즘이 최단 경로 계산에 필요한 대부분의 데이터가 미리 정의되어 있기 때문에 정적 환경 및/또는 전역 경로 계획에 가장 적합하다는 것을 일반화할 수 있지만, 다익스트라 알고리즘이 동적 환경에 사용된 응용 프로그램이 있다. 이 경우 환경은 부분적으로 알려져 있거나 완전히 알려져 있지 않으므로 장애물에 대한 노드 정보가 즉시 계산된다. 이를 로컬 계획이라고 하며, 다익스트라 알고리즘은 최단 경로 계산의 평가를 위해 실행된다. 그러나 동적 환경 내에서 다익스트라 알고리즘만 사용할 수 없다[128].



4.3.3. A* 알고리즘

자율 주행차의 경로 계획에 흔히 사용되는 또 다른 알고리즘으로 A*가 있다. A*는 휴리스틱 기반의 탐색 알고리즘이다. 너비 우선 탐색(breadth first search, BFS)이나 깊이 우선 탐색(depth first search, DFS)과 마찬가지로, A*도 몇 가지 측정 기준에 따라 공간을 탐색한다. A*는 ‘가치 기반(merit-based)’ 또는 ‘최적 우선(best-first)’ 탐색 알고리즘으로 볼 수 있다.

A*는 openSet이라는 노드 집합을 이용하는데, 이 집합은 탐색할 잠재 노드를 담는다. A*는 루프를 돌 때마다 탐색할 최소 비용 노드를 하나씩 뽑는다. 뽑은 노드의 비용 $f(v)$ 는 $g(v)$ 와 $h(v)$ 라는 2개의 항을 더한 값이다. A*의 탐색 트리에 있는 각 노드는 출발 노드에서 자신에 이르는 비용을 갖는데 이 값이 $g(v)$ 이다. 반면 $h(v)$ 는 모든 노드마다 갖는 휴리스틱 비용이다. 휴리스틱 비용 $h(v)$ 는 현재 노드에서 목적 노드에 이르는 최소 비용 추정치를 나타낸다. 휴리스틱 비용 $h(v)$ 가 특정 속성을 만족할 때 A*는 출발지에서 목적지에 이르는 최소 비용 경로를 찾을 수 있다. A* 탐색 과정에서 루프를 돌 때마다 목적 노드가 확장될 때까지 최소 비용이 $f(v) = g(v) + h(v)$ 인 노드가 계속 확장된다.

이 알고리즘은 휴리스틱 기반의 탐색 알고리즘이기 때문에 $h()$ 가 허용(admissible) 속성[129]을 만족한다면, 즉 최소 비용 추정치 $h(v, dst)$ 가 실제 최소 비용을 절대로 초과하지 않는다면, A* 알고리즘은 항상 최소 비용 경로를 찾을 수 있다. 반면 이 속성을 만족하지 않으면 최소 경로를 찾지 못할 수도 있다. 자율 주행 자동차의 경로 계획에서 차로 지점들이 연결된 그래프를 통해 주어진 2개의 차로 지점 A와 B 사이의 휴리스틱 척도를 정의하는 한 가지 방법은 $h(u, v) = dist(u, v)$ 이다. 여기서 $dist()$ 는 지리 좌표계에서 두 차로 지점 사이의 메르카토르(Mercator) 거리를 나타낸다[115]. A*는 데이크스트라 알고리즘을 확장한 최적 우선 탐색 알고리즘이라고 볼 수 있다. 마찬가지로 데이크스트라 알고리즘도 $h(u, v) = 0$ 인 특수한 경우에 대한 A* 알고리즘이라 볼 수도 있다.

중요한 것은 A* 알고리즘이 계산적으로 효율적이라는 것이다[130, 131]. 따라서 정적 환경에 배포된 애플리케이션에 적합하다. A*와 그 변형들의 계산 속도와 효율성은 휴리스틱 함수의 정확성에 달려있다.

4.3.4. D* 알고리즘

부분적으로 알려져 있고 효율적인 방식으로 동적 환경에서 경로 계획을 세우는 것이 자동화 차량의 경우 점점 더 중요하다. 이 문제를 해결하기 위해, D*(또는 Dynamic A*) 알고리즘은 움직이는 장애물 중 충돌 없는 경로를 생성하는 데 사용



된다. D*는 비용 지도와 이전에 계산된 비용 지도를 부분적으로 복구하는 정보 증분 검색 알고리즘이다.

D* 알고리즘은 오픈 리스트에서 제거될 때까지 차량의 상태를 처리하고, 동시에 백 포인터와 함께 상태 시퀀스를 계산하여 차량을 목표 위치로 유도하거나 감지된 장애물로 인한 비용을 업데이트하고 해당 상태를 오픈 리스트에 배치한다. 개방형 목록의 상태는 현재 상태에서 목표까지의 경로 비용이 최소 임계값보다 작을 때까지 처리되고, 비용 변경은 다음 상태로 전파되며, 차량은 목표를 향해 새로운 시퀀스의 백 포인터를 계속 추적한다[132]. D*는 최적 재계획자(re-planner)보다 200배 이상 빠르다[133, 134]. D* 알고리즘의 주요 단점은 다른 D* 변형과 비교할 때 높은 메모리 소비량이다[135].

D* Lite 알고리즘[136]은 시작 정점과 목표 정점을 교환하고 모든 에지를 알고리즘적으로 역전시킴으로써 LPA*(Lifelong Planning A*)를 기반으로 한다. 알고리즘은 계산된 rhs값(right-hand side value)을 최소화하여 미지의 동적 환경에서 목표 노드에서 시작 노드로 가는 최단 경로를 찾는다. 정점의 주요 값은 연결 에지 가중치의 변화에 따라 연결이 변경될 때 계산되고 업데이트된다. 이러한 변경으로 휴리스틱은 목표에서 원래 시작까지의 예상 비용에서 목표 노드에서 새 시작 노드로의 예상 비용으로 업데이트된다.

이 방법을 사용하면 연결이 변경될 때마다 우선 순위 큐를 통과하지 않는다. 따라서, D*Lite 알고리즘은 우선순위 큐의 순서를 바꾸지 않기 때문에 LPA*에 비해 계산 비용이 낮다. 이 알고리즘은 예상치 못한 장애물이 발생할 때 알고리즘이 신속한 재계획 결과에 도달할 수 있는 복잡한 환경에서 자율 주행 차량의 경로 계획에 적합하다[137].

향상된 D* Lite 알고리즘[138]은 실제로 D* Lite 알고리즘을 개선한 것으로, 동일한 경로 찾기 원칙을 유지한다. 하지만 복잡한 장애물을 피하고, 로봇이 두 장애물 사이를 가로지르고, 장애물의 날카로운 모서리를 가로지르는 것을 방지하며, 필요하다면 가상 벽을 만들고, 목표 위치에서 시작 위치로 가는 최단 경로를 생성하는 불필요한 경로를 제거하는 것을 강조하는 개선을 통해 문제를 극복한다. 이 알고리즘은 장애물을 탐지하기 위한 소나 센서가 장착된 팀 아미고봇을 사용하여 실제 시스템에서 구현되었다[138].

D* 알고리즘과 그 변형은 목표에 대한 최적의 경로를 찾는 동안 경로 비용이 변경되는 모든 경로 비용 최적화 문제에 사용될 수 있다. D*는 이러한 변경 사항이 검색 공간의 현재 노드에 가깝게 감지될 때 가장 효율적이다. D* 알고리즘은 행성 탐사선 임무 계획을 포함하여 광범위한 응용 분야를 가지고 있다[139].

4.3.5. RRT(Rapidly-Exploring Random Trees) 알고리즘



A*와 같은 알고리즘은 본질적으로 정적이며 사전에 지정된 경로를 필요로 한다. 이제 경로를 미리 지정할 필요가 없는 RRT[140, 141]와 같은 동적 알고리즘에 대해 논의해 보자. RRT 알고리즘은 모든 영역에서 확장되며, 각 노드에 할당된 가중치를 기반으로 시작에서 목표까지의 경로를 생성한다. RRT는 광범위한 경로 계획 문제를 처리하기 위해 도입되었다. 그것들은 비홀로노믹 제약 조건(위치 제약 조건에 통합할 수 없는 제약 조건)을 처리하도록 특별히 설계되었다.

RRT와 확률론적 도로 지도(PRM)는 동일한 바람직한 특성을 공유하며, 둘 다 휴리스틱과 임의의 매개 변수를 사용하지 않고 설계되었다. 따라서 결과에서 더 나은 성능과 일관성이 제공된다. PRM은 솔루션을 찾기 위해 수천 개의 구성 또는 상태의 연결을 필요로 하는 반면, RRT는 솔루션을 찾기 위해 상태 간에 어떠한 연결도 요구하지 않는다. 이는 RRT를 비홀로노믹 및 운동학적 계획에 적용하는 데 도움이 된다[142].

RRT는 공간을 빠르게 샘플링하여 확장하고, 시작점부터 성장하며, 트리가 목표점에 충분히 근접할 때까지 확장한다. 모든 반복에서 트리는 무작위로 생성된 꼭짓점의 가장 가까운 꼭짓점으로 확장된다. 이 가장 가까운 꼭짓점은 거리 메트릭의 관점에서 선택된다. 여기에 적용되는 메트릭은 유클리드, 맨해튼 또는 기타 거리 메트릭일 수 있다.

RRT는 이미 탐색된 장소에서 크게 확장되는 경향이 있는 기본 랜덤 트리에 비해 차량 구성 공간의 미탐색 부분에서 크게 확장된다(그림 4.4 참조). 따라서 RRT는 미개척 지역에 편중되어 있다고 할 수 있다. RRT의 꼭짓점은 균일한 분포를 따른다. 알고리즘은 비교적 간단하며 RRT는 에지 수가 최소일지라도 항상 연결되어 있다.

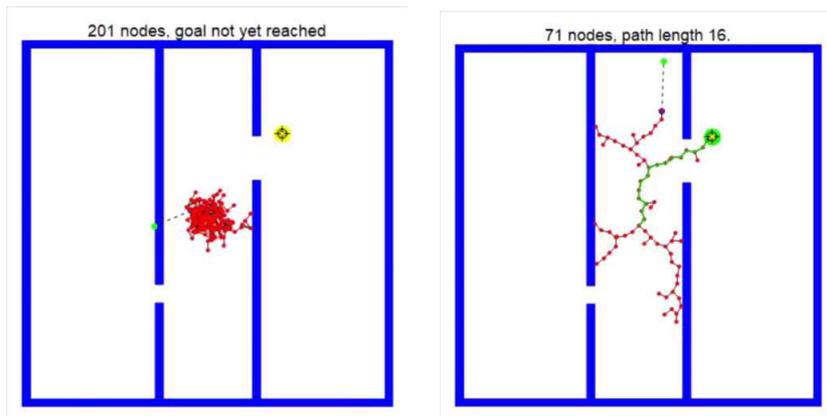


그림 4.4. (a) 기본 랜덤 트리의 탐색, (b) RRT의 탐색(출처: [143])

기본 랜덤 트리의 탐색은 트리에서 무작위로 노드를 선택하고 랜덤 방향으로 에지를 추가한다. RRT의 탐색은 고차원 공간에서 경로를 효율적으로 탐색하도록



설계되었다[143].

RRT 알고리즘은 비홀로노믹 제약에 대처할 수 있기 때문에 거의 모든 휠 시스템에 적용할 수 있다. 사용자의 요구에 따라, 그들은 RRT의 변형을 선택할 수 있다[144].

4.3.6. 유전 알고리즘

그리드 기반 알고리즘과 잠재적 필드와 같은 이산 경로 계획 알고리즘은 상당한 양의 CPU 성능과 메모리가 필요하다. 이 절에서는 이러한 한계를 극복하는 데 도움이 되는 유전자 알고리즘(GA)을 소개한다. 예를 들어, 유전 알고리즘은 넓은 검색 공간을 커버하고 최소한의 메모리와 CPU 자원을 사용한다는 장점으로 적용될 수 있다. 그들은 또한 변화하는 환경에 적응할 수 있다.

한 가지 단점은 최적화 문제에 대해 발견된 솔루션이 항상 전역 최소값(예: 전체 최단 경로)이 아닐 수 있다는 것이다. 유전 알고리즘의 흥미로운 응용 중 하나는 축구를 할 수 있는 작은 크기의 휴머노이드 로봇이다[145]. 로봇이 골대 안으로 공을 차 넣으려면 충돌을 피하고 골대 쪽으로 향해야 하기 때문에 상대 팀이 장애물이 되어 공을 차려면 로봇이 골대를 알아야 한다는 점에 주목한다. 따라서 로봇의 경우 동적 경로 계획이 필요하다.

동적 환경에도 알고리즘이 적용되기 때문에 두 지점 간 충돌 없는 경로를 찾는 최적 솔루션은 주변 변화에 따라 지속적으로 업데이트해야 한다. 이 경우 진화적 방법은 최적의 해결책을 수렴한다. 가능한 모든 솔루션은 각 유전자가 매개 변수를 나타내는 모집단의 객체로 표현된다. 개인은 완전한 유전자에 의해 형성된다.

부모 세대 중에서 가장 우수한 개인을 선발하고 교차, 돌연변이 등 유전자 조작자를 적용해 추가적인 해결책을 모색하는 방식으로 새로운 세대가 형성된다. 새로운 세대의 각 자손들은 그 문제를 위해 고안된 피트니스 기능으로 테스트를 받는다. 모든 자손 중에서, 가장 좋은 객체는 다음 세대의 부모로 선택된다.

피트니스 기능은 시뮬레이션을 최적의 설계 솔루션으로 안내하는 데 사용된다. 피트니스 기능은 동적 경로 계획을 위해 최단 경로뿐만 아니라 부드러움과 간극을 고려해야 한다[146].

유전자 조작자는 선택된 부모를 위해 그럴듯한 경로를 진화시키는 데 사용된다. 선택 조작자는 가장 적합한 개인을 선택하고 다음 세대에 유전자를 물려주도록 합니다. 교차 연산자는 짝짓기를 할 각 부모 쌍에 대해 유전자 중에서 무작위로 선택된 교차점을 선택한다. 변환 연산자는 다양성을 유지하기 위해 비트 문자열에서 일부 비트의 플립을 수행한다. GA는 최적의 경로를 생성하고 GA의 최적화 능력을 활용한다[147]. 알-타하르 등은 비동적 환경에서 경로 계획 문제를 해결하기



위해 GA 접근 방식을 사용하는 방법을 설명했다[148].

그들의 결과는 채용된 방법이 다른 정적 분야에서 경로 계획 문제를 처리하는데 효율적이라는 것을 보여주었다. 이러한 종류의 문제를 해결하는 데 GA를 적합하게 만든 중요한 특성은 GA가 본질적으로 병렬 검색 방법이며 주어진 환경에서 최적의 경로를 검색할 수 있는 능력을 가지고 있다는 것이다[147, 149]. 이것들은 U자형 장애물이 있는 복잡한 환경에서 경로 계획에 유용하다[132].

4.3.7. 개미 집단 알고리즘

컴퓨터 과학자와 생물학자들은 경로 계획 최적화 알고리즘의 설계를 위해 자연으로부터 영감을 받았다. 가장 짧고 충돌 없는 경로를 찾기 위해 개미들의 집단 행동에서 영감을 받은 휴리스틱 접근 방식을 기반으로 하는 개미 군집 최적화(ACO) 알고리즘이 그러한 파생 알고리즘 중 하나이다. 이 알고리즘은 마르코 도리고(Marco Dorigo)가 1992년 박사학위 논문 “개미 시스템: 협력 에이전트의 콜로니(colony of cooperating agents)”에서 개미 시스템(AS) 이론에서 먹이를 찾는 개미를 시뮬레이션하기 위해 처음 제안했다.

문제의 다양한 특성에 따라 최단 경로를 찾기 위한 확률론적 휴리스틱 알고리즘인 다양한 알고리즘이 도출되었다. 개미들이 음식을 찾을 때, 그들은 화학물질인 페로몬이라고 불리는 물질을 길을 따라 방출한다. 이어서 다음 개미들은 이전 개미들이 남긴 페로몬 농도에 따라 적합한 경로를 선택할 것이며, 어떤 개미라도 그 경로를 선택할 확률은 동일하다. 기존 페로몬의 농도가 높을수록 개미가 경로를 선택할 확률도 높아진다.

경로 탐색 횟수가 늘어나면 짧은 경로일수록 이 경로를 방문한 개미가 많아 페로몬 농도가 높아지는 반면, 다른 경로의 경우 방문하는 개미가 적어 페로몬의 자연증발 현상이 나타나기 때문에 페로몬 농도가 낮아진다. 따라서, 개미는 페로몬 정보에 따라 검색하여 더 짧은 거리를 가진 경로로 이동할 수 있다[150].

ACO 알고리즘의 두드러진 특징은 목표를 신속하게 발견하는 데 도움이 되는 긍정적인 피드백, 조기 수렴을 피하는 분산 계산, 초기 단계에서 목표를 찾는 데 도움이 되는 탐욕스러운 휴리스틱이다. 경로에 더 많은 페로몬이 있을수록, 개미가 그 길을 택할 확률은 더 높아진다. ACO 알고리즘은 로봇 동적 경로 계획에 사용되며 차량 라우팅 문제에 대해 사전 최적화된 경로를 찾는 데 활용된다[151]. 기존의 개미 군집 알고리즘은 휴리스틱 검색의 한계로 인해 수렴 속도가 느리다. 차량 경로 계획 적용 시 최적화 초기 단계에서 국지적 최적 솔루션에 빠지기 쉬우므로 정체 현상이 발생한다.

향상된 개미 군체 알고리즘(EACA) [152]은 개미가 상태 전이 확률 함수를 사용



하여 먹이를 찾고 등지로 돌아오는 과정에서 무작위로 움직이기 때문에 전통적인 개미 집단 알고리즘의 단점을 극복한다. 영감 기능을 수정한 목적은 현재 노드에서 대상 노드에 대한 개미의 인지를 향상시키는 것으로, 이를 통해 탐색 시간을 단축하고 국소 최적 상태에 빠지는 것을 방지할 수 있다[153]. 이 알고리즘은 수렴 속도를 높이고 복잡한 시나리오에서도 합리적인 솔루션을 효과적으로 생성할 수 있으며, 이는 ACO 알고리즘의 몇 가지 개선 사항과 함께 충돌 회피 시스템에 이 알고리즘을 사용하도록 영감을 주었다[154-156].

개선된 개미 집단 최적화 알고리즘 [157]은 대상 탐색 프로세스의 각 사이클 후에 페로몬 업데이트 방법을 실행하고, 개선된 비용 함수를 이용하여 최적 경로를 계획한다. 이 알고리즘은 ACO의 계산 능력을 향상시키고 최적 경로로 훨씬 빠르게 수렴한다. 동적 환경에서 다중 AGV(automated guided vehicles)에 대한 대규모 차량 배치 문제와 경로 계획을 해결하는 데 사용된다[53]. 동적 환경에서의 효과와 컴퓨팅 파워의 보다 효율적인 사용이 가능하기 때문에 무인항공기(UAV)의 최적 경로 계획에 사용된다[158].

Adaptive Ant Colony (AAC) 알고리즘 [159]은 증발률의 값을 적응적으로 업데이트하고 각 경로에서 페로몬을 업데이트하여 ACO 알고리즘을 개선한다.

적응 개미 집단(AAC) 알고리즘은 ACO 알고리즘에서 조기 및 정체 결함을 극복하는 전역 최적 경로, 더 나은 안정성, 수렴 및 작동 속도를 갖는다. 차량 경로 계획 문제를 해결하기 위해 AAC는 알고리즘이 두 번 반복 실행되는 적응형 메모리 프로그래밍 알고리즘(AMP)과 함께 사용되어 왔다. 첫 번째 반복 AMP는 AAC가 국소 최소로 떨어지는 단점을 커버할 수 있도록 실행되며, AMP의 결과, 즉 두 번째 반복 AAC가 실행된다. 이를 통해, AAC는 물류에서 사용되는 차량 경로 계획에 적용했을 때 경로 길이를 최소화할 수 있다[160].

ACO 알고리즘과 그 변형은 병렬 최적화 알고리즘의 유형으로, 좋은 수렴 및 전역 최적화 성능을 가지고 있다. 적응 개미 군집 알고리즘은 우수한 전역 최적화 능력을 가지고 있으며, ACO 알고리즘의 다른 변형에 비해 더 나은 최적화 성능을 보인다.

4.3.8. 경로 계획 그래프 비용

실제로 자율 주행차의 경로 계획을 구현할 때는 알고리즘의 종류보다 비용 설정이 더 중요하다. 경로 계획 모듈에서 차로 지점 사이의 비용을 설정하는 방법이 대단히 중요하다. 예를 들어 동적으로 받은 교통 정보를 통해 현재 도로가 매우 붐빈다는 사실을 알면 이 도로에 속한 차로 지점을 연결하는 에지의 비용을 높게 설정하는 방식으로 경로 계획 모듈이 혼잡한 차로를 피해서 경로를 찾게 만



들 수 있다. 마찬가지로 특정 도로에 대한 교통 통제 상황이 발생했다면, 그 도로에 있는 차로 지점을 연결하는 에지의 비용을 (무한대와 같이) 충분히 높게 설정해서 탐색 알고리즘이 이러한 차로를 선호하지 않거나 피하게 만들 수도 있다. 또한 다른 차로보다 특정한 차로를 선호하도록 차로 지점 사이의 비용을 동적으로 조절할 수도 있다. 실제 도로 그래프 데이터가 매우 크다는 사실을 감안하면, 경로 계획 모듈은 이러한 도로 그래프를 미리 읽어들이는 상태에서 차로 지점 그래프를 동적으로 구성할 수도 있다. 주어진 반경의 도로 그래프에 목적지에 이르는 경로를 못 찾으면, 더 큰 반경의 도로 그래프 데이터를 다시 읽거나, 차로 지점 그래프를 재구성하는 방식으로 경로를 다시 계산한다.

경로 계획 요청은 크게 두 종류가 있다. 하나는 자율 주행 자동차가 주행을 시작할 때 승객이 경로 계획 요청을 보내는 방식으로 출발지와 목적지를 설정하는 것이다. 다른 하나는 (동작 결정 모듈, 모션 계획 모듈 같은) 다운스트림 모듈에서 설정하는 것이다. 이 과정을 엄격한 경로 계획(strong routing)과 유연한 경로 계획(weak routing)이란 개념으로 구분할 수 있다. 엄격한 경로 계획은 다운스트림 모듈이 경로 계획 모듈의 결과를 그대로 따르는 방식이다. 이를 위해 결정 및 계획 모듈은 경로 계획의 결과에서 지정한 차로를 최대한 따른다. 경로 계획의 결과에서 지정한 차로를 똑같이 따르는 것이 도저히 불가능할 때는 경로 계획 요청의 두 번째 종류에서 설명한 것처럼 다운스트림 모듈에서 경로 계획 요청을 다시 보낸다. 유연한 경로 계획은 다운스트림 모듈이 상황에 따라 경로 계획 모듈의 결과를 그대로 따르지 않는다. 경로 계획 모듈의 결과에서 지정한 것과 다른 차로 시퀀스를 이용할 수도 있다. 다시 말해, 자율 주행 자동차의 동작이 달라질 수 있다. 예를 들어 비교해보자. 경로 계획 모듈의 출력에서는 현재 차로를 유지하라고 나왔는데 현재 차선의 전방에 굉장히 느리게 달리는 차가 있다고 하자. 엄격한 경로 계획을 적용하면 주행 속도를 줄이고 앞에서 천천히 달리는 자동차를 그대로 따라간다. 반면 유연한 경로 계획을 적용하면 대부분의 운전자들이 하듯이 자율 주행 자동차도 차로를 변경해 서행하는 자동차를 추월한 뒤 이전 차로로 복귀한다. 엄격한 경로 계획과 유연한 경로 계획 중 어느 방식을 적용하더라도, 사고가 발생하거나 급히 피하는 동작을 수행할 때 다운스트림 모듈은 항상 안전을 최우선으로 작동한다. 이런 경우에는 주로 경로 계획을 다시 요청한다.

4.4. 결론

4장에서는 예측 및 경로 계획 모듈에 대해 살펴보았다. 여기서 소개한 예측 및 경로 계획 모듈은 기존의 계획 및 제어 모듈과 좀 다르다. 여기서 제안한 넓은 의미의 계획 및 제어 프레임워크의 관점에서 보면 여기서 소개한 예측 및 경로 계획



모듈은 기존의 모션 계획 모듈에 대한 입력 값을 생성한다. 따라서 예측 및 경로 계획 모듈을 넓은 의미의 계획 및 제어 프레임워크의 일부로 봤다. 트래픽 예측 모듈은 동작 분류와 궤적 생성이라는 두 가지 문제로 추상화하며 4장에서는 자율주행 자동차가 경로 계획 모듈의 결과로 주어진 차로 시퀀스를 따라 목적지까지 주행하는, 경로 계획 기법을 소개했다. 다음 장에서는 이 장에서 예측한 궤적과 경로를 바탕으로 동작 결정, 모션 계획 피드백 제어 등과 같은, 기존에 익숙한 계획 및 제어 모듈에 대해 설명한다.

5. 결정, 계획, 제어

5.1. 동작 결정

경로 계획을 세운 다음에 자율 주행차는 도로규칙과 운전 관행에 따라 다른 교통 참여자들과 상호작용하면서 선택한 경로를 운행할 수 있어야 한다. 선택한 경로를 규정한 일련의 도로가 주어진 상태에서 행동 플래너는 다른 교통 참여자들의 행동, 도로조건 및 인프라로부터의 신호 등과 같은 인식을 기반으로 임의의 점에서 적절한 주행을 선택할 책임이 있다.

예로 자율 주행차가 교차점 직전의 정지선에 도달했을 때, 행동 플래너는 자동차가 정지하도록 명령하고, 교차점에 있는 다른 자동차나 바이크, 보행자들의 행동을 관측하고 자동차가 주행으로 전환하도록 처리해야 할 것이다.

실제 운전은 특히 도심환경에서 다른 교통 참여자들의 의도에 관한 불확실성에 의해 특정된다. 다른 자동차나 바이크, 보행자들의 의도 예측이나 다른 자동차들의 미래 궤적에 관한 추정의 문제들 또한 연구되어 왔다. 제안된 해결 방법들 가운데 가우시안 혼합모델(Gaussian Mixture Model: GMM)[161]과 같은 기계학습 기반 기술도 있다.

다른 교통 참여자들의 행동에서의 불확실성(GMM으로 생성된 것)은 마르코프 결정과정(Markov decision processes: MDP)과 같은 확률적 계획 형식론의 사용으로 결정을 위한 행동계층에서 공통적으로 고려되었다. 가령 부분적 관측 가능 마르코프 결정과정(POMDP) 구성은 관측되지 않은 운전 시나리오와 보행자 의도를 명시적으로 모델화하는데 응용될 수 있고, 구체적인 근사적 해결책을 생성한다[162].

특히 POMDP는 MDP를 일반화시킨 것이다. POMDP는 에이전트 결정과정을 모델화한 것이고, 시스템 다이내믹스가 MDP에 의해 결정되지만 에이전트는 기본상태를 직접 관측할 수 없는 것으로 가정한다. 대신에 이는 일련의 관측과 관측 확률 그리고 기본 MDP 등에 의한 가능한 상태들에 걸쳐 확률분포를 유지해야 한다.

5.1.1. 마르코프 결정 과정

결정에 직면할 때 선택할 수 있는 많은 수의 다른 동작들이 될 수 있고 서로 다른 결과를 가져올 수 있다. 최상의 동작을 선택하는 것은 동작의 직접적인 영향 그 이상에 관한 사고를 요구한다. 직접적인 영향은 확인하기 쉽다. 그러나 장기간의 영향은 항상 분명하지 않다. 따라서 불충분한 직접적인 영향의 동작도 더 좋은 장기간의 결과를 가질 수 있을 것이다.

마르코프 결정 과정(Markov Decision Process, MDP)은 자동화할 수 있도록 결정과정을 모델화하는데 사용될 수 있다. 의사결정과정의 공식화에 MDP 사용으로 의



사결정 문제를 자동으로 해결하는데 많은 수의 알고리즘을 사용할 수 있다. MDP는 다음과 같이 다섯 가지 요소로 구성된 튜플 (S, A, P_a, R_a, γ) 로 정의한다.

- S 는 자율 주행차의 상태 공간을 나타낸다. 상태 공간의 분할은 자율 주행 자동차의 위치와 지도 요소 모두를 고려해야 한다. 위치는 자율 주행차 주변의 정사각형 영역을 고정된 길이와 너비를 가진 그리드 단위로 분할해서 표현할 수 있다. 다양한 도로 지도 객체를 고려하면 현재 자율 주행차가 있는 차로와 옆 차로를 비롯한 지도 객체를 다양한 방식으로 조합해서 공간을 생성할 수도 있다.

- A 는 동작 결정 출력 공간을 표현하며 이 공간은 자율 주행차가 수행할 수 있는 모든 동작을 고정된 집합으로 표현한 것이다. 결정 상태의 예로 현재 차로에서 전방에 달리는 자동차를 추종하는 상태(Follow) 옆 차로로 차로 변경하는 상태(Switch Lane) 교차로에서 좌/우회전하는 상태(Turn Left/Right) 다른 자동차에게 양보하는 상태(Yield) 또는 추월하는 상태(Overtake) 신호등과 보행자 앞에서 정지하는 상태(Stop) 등이 있다.

- $P_a(s, s') = P(s' | s, a)$ 는 자율 주행차가 현재 상태 s 에서 동작 a 를 수행할 때 상태 s' 에 도달할 조건부 확률을 나타낸다.

- $R_a(s, s')$ 는 보상 함수(reward function)로서 상태 s 에서 동작 a 를 수행함으로써 상태 s' 으로 변환할 때의 보상을 나타낸다. 여기서 보상이란 이러한 상태 변환을 종합적으로 평가하는 척도다. 보상에서 고려해서 표현할 요소는 안전성, 편안함, 목적지 도달, 수행할 다운스트림 모션 계획에 대한 난이도 등이 있다.

- γ 는 보상에 대한 감쇠 요소(decay factor)다. 현재 시간에서 보상은 인자 1을 갖고, 다음 시간 프레임에 대한 보상은 인자 γ 에 의해 감소한다. 따라서 미래의 시간 프레임 t 에 대한 보상은 인자 γ 만큼 줄어든다. 감쇠 요소는 같은 양의 보상에 대해 항상 미래보다 현재에 좀 더 가치가 있도록 보장한다.

전형적인 MDP에서 동작 결정 모듈이 해결할 문제는 $\pi: S \rightarrow A$ 로 표현되는 최적 정책(policy)을 찾는 것이다. 정책은 주어진 상태 s 에 대한 동작 결정 $a = \pi(s)$ 를 계산한다. 정책이 결정되면 전체적인 MDP는 마르코프 체인(Markov Chain)으로 볼 수 있다. 동작 결정 정책 π 는 현재 시간부터 미래까지의 누적 보상을 최적화한 것이다. 여기서 주의할 점은 보상이 결정적이 아니라 확률 변수이면 정책은 기대 누적 보상을 최적화한다는 것이다.

MDP에 대한 해를 방침(policy)이라 부르고, 이는 단순히 각 상태가 취할 최상의 동작을 규정한다. 하나의 방침을 유도하기 위해서는 가치함수(value function)를 최적화하는 것이 필요하다. 가치함수는 각 상태에 수치를 규정한다. 따라서 MDP 사용에서 상태집합, 선택할 동작 집합, 보상 함수 그리고 확률적 천이 행렬을 갖는다. 목적은 상태에서 동작으로의 매핑을 유도하는 것인데, 이는 각 상태가 취할



최상의 동작을 나타낸다.

여기서는 보상 함수 $Ra(s, s')$ 을 설계할 때 고려할 사항들을 중점적으로 소개한다. MDP 시스템을 구축할 때 고려해야 할 대단히 중요한 사항이기 때문이다. MDP 기반의 결정 모듈에서 보상 함수를 잘 만들려면 다음과 같은 사항을 고려한다.

- 목적지 도달: 자율 주행 자동차가 목적지에 도달하기 위해 경로 계획 모듈의 출력 값 경로를 따르도록 권장한다. 좀 더 자세하게 말하면, 정책에 따라 선택한 동작 $\alpha = \pi(s)$ 를 선택했을 때, 자율 주행 자동차가 경로에서 이탈하면 패널티를 부과해야 한다. 이와 반대로 주어진 경로를 잘 따라가면 보상을 지급한다.
- 안전성 및 충돌방지: 상태 공간을 자율 주행 자동차를 중심으로 $N \times N$ 정사각 그리드로 표현했다면 충돌을 발생시키는 그리드로 이동하는 모든 결정에 패널티를 부과해야 한다. 충돌 가능성이 낮거나 충돌 가능 지점과 거리가 멀수록 보상을 지급한다.
- 편안함(comfortableness) 및 부드러움(smoothness): 두 가지 요소는 서로 밀접한 관련이 있다. 편안한 주행은 대부분 급격한 동작을 거의 또는 전혀 하지 않는다. 그리고 급격한 동작이 없다면 다운스트림 모듈이 내리는 결정도 대부분 부드럽게 동작한다. 예를 들어, 비슷한 속도를 유지하며 가속 또는 감속하는 동작이 급격한 가속이나 감속을 하는 동작보다 더 높은 보상을 받아야 한다.

5.1.2. 가치반복(Value Iteration) 알고리즘

가치순환은 최적 MDP 방침과 그 값을 계산하는 방법이다. 가치 반복 알고리즘은 이전의 것에서 도출되는 일련의 가치함수를 구해서 이를 계산한다.

가치반복 알고리즘은 1의 수평길이에 대한 가치함수를 구하는 것으로 시작한다. 이는 단일 결정이 필요하다는 것이 주어진 조건에서 개별 상태의 값이 될 것이다. MDP에서 이를 구하기 위해 해야 할 것이 많지 않다. 보상을 갖고 있음을 기억하면, 이는 각 상태에서 개별 동작이 얼마나 양호한 것인지를 규정한다. 수평길이가 1 이기 때문에 단지 보상을 볼 수 있고, 각 상태에 대한 가장 높은 보상 값을 갖는 동작을 선택할 수 있다.

알고리즘의 2번째 반복인 다음 단계는 수평길이가 2인 경우 대한 가치함수를 결정하는 것이다. 2단계가 나아갔을 때 동작의 가치는 취하게 될 즉시 동작(immediate action)에 대한 보상을 선택한 다음 동작의 가치를 더한 것이다. 수평길이가 1에 대한 각 상태의 값을 이미 계산했다. 그래서 수평길이가 2에 대한 값을 구하기 위해서는 2가지 결정이 이루어질 것이 주어진 상태에서 최상의 값을 갖는 행동을 구하기 위해 이미 계산된 가치함수에 가능한 각 행동의 보상 값을 더할

수 있는 것이다.

알고리즘을 다시 반복시킨다. 수평길이 2의 가치함수를 사용해서 수평길이 3의 가치함수를 구한다. 이러한 처리를 원하는 수평선(horizon)에 대한 가치함수를 구할 때까지 반복한다.

5.1.3. 부분적으로 관측가능한 MDP(Markov Decision Process)

MDP와 POMDP간의 주요 차이는 과정의 현 상태를 관측할 수 있느냐의 여부에 있다. POMDP에서 일련의 관측을 모델에 더한다. 따라서 현 상태를 직접 관측하는 대신에 상태가 무엇인가에 관한 힌트를 제공하는 관측을 준다.

관측은 관측 모델을 규정하는 것이 필요한 확률적이 될 수 있음을 주목하라. 이 관측 모델은 단지 모델에서 각 상태에 대한 개별 관측의 확률을 이야기할 뿐이다. 현 상태에 대한 직접적인 접근을 갖고 있지 않으므로, 과정의 전체 히스토리의 트랙을 유지하는 것이 요구된다. 특히 주어진 시간의 한 점에서 히스토리는 시작상황, 수행한 모든 동작 및 관측에 관한 지식으로 구성된다.

모든 상태에 걸쳐 확률분포의 유지는 완전한 히스토리를 유지하고 있는 것처럼 동일한 정보를 제공한다. POMDP에서 상태에 걸쳐 이 확률분포를 유지해야 한다. 동작을 수행하고, 관측할 때 확률분포함수를 갱신한다. 분포함수의 갱신은 천이 및 관측 확률의 사용을 포함하고 있다.

Ulbrich 등은 도시 환경에서 주행 중 차선 변경을 수행하기 위해 온라인 POMDP를 행동 선택에 적용한다[162]. POMDP의 복잡성을 줄이고 실시간 의사결정을 할 수 있도록 제안된 알고리즘을 두 단계로 나누었다. 첫 번째 단계에서, 상황은 신호 처리 네트워크에 의해 평가된다. 이 네트워크는 차량 주변의 물체와의 충돌까지 상대적인 거리, 속도, 시간을 고려하여 차선 변경 가능 여부를 출력하는 그래프이다. 네트워크 출력으로, POMDP 의사 결정 알고리즘은 현재 신념 상태에 대해서만 온라인으로 계획을 세운다. 이것은 현재 상태에서 도달할 수 있는 믿음 상태만 검색하여 만들어진다. 따라서 POMDP 모델은 8개의 상태만 있으면 된다. 알고리즘은 높은 수준의 의사 결정 품질을 유지하면서 의사 결정 일관성을 보여주는 실제 도심 트래픽에서 평가되었다.

Brechtel 등은 도로 참여자의 상호작용을 고려하여 잠재적으로 숨겨진 물체와 관찰 불확실성에 대해 추론하기 위해 연속적인 POMDP 접근법을 사용했다[163]. 그들은 계획의 속도를 높이기 위해 유한한 일련의 정책을 가정한다는 생각을 사용했다. 첫 번째 단계에서 보상 기능은 목표 영역에 도달하기 위한 가속 및 감속 비용을 반환함으로써 편안함과 효율성을 최적화하는 것을 목표로 한다. 이 단계는 차량의 상태와 이전에 정의된 목표에 따라 달라진다. 두 번째 단계에서는 다른 도



로 사용자와의 충돌에 대한 더 높은 비용을 추가하여 다른 교통 참여자를 고려한다. 두 단계 비용은 주행 목표를 나타내는 스칼라 값을 갖는 보상 함수로 요약된다. 그들의 접근 방식은 병합 시나리오에서 평가되었다. 이 시나리오에 대한 정책 및 가능한 모든 결과는 오프라인에서 사전 계산되었다.

Galceran 등은 차량 거동과 인근 차량을 별개의 정책 집합으로 모델링하는 통합 추론 및 행동 선택 접근법을 제안했다[164]. 이 작업에서, 그들은 차선과 교차로 주행 상황에 대한 일련의 수공학적 정책을 사용했다. 그들은 다른 차량의 역사에 대한 베이지안 변경점 탐지를 사용하여 가능한 미래 조치를 추론하고, 각 인근 차량이 실행할 수 있는 잠재적 정책에 대한 분포를 추정했다. 그런 다음 행동 선택 알고리즘은 포워드 시뮬레이션을 통해 예측 차량의 정책을 평가하는 POMDP 솔루션을 근사화하여 최대 보상값으로 정책을 실행한다. 그러나 그들은 대다수의 운전 참가자들이 교통 규칙을 따라 규칙적이고 예측 가능한 방식으로 행동한다고 가정한다. 실험은 자율주행 자동차 플랫폼을 사용하여 수행되었다.

Wray 등은 상호 작용이 있는 여러 온라인 의사 결정 구성 요소(MODIA, multiple online decision-components with interacting actions)를 사용하는 자율 주행 차량에 대한 교차 행동 선택을 제안했다[165]. MODIA는 또한 트래픽 참가자를 고려하지만 별도의 개별 MDP로 모델링한다. 각 민주당은 추정된 일련의 조치를 생성하면서 각 단계마다 취해야 할 자신의 신념과 제안된 조치를 유지한다. 사전 편집 실행자 동작 함수(LEAP)는 이 집합에서 최상의(기본 설정 측면에서) 동작만 실행합니다(예: 중지 동작은 기본 설정을 가집니다). 동작은 단순히 정지, 예지 또는 고이며 자율 주행차의 궤적을 따라 원하는 속도와 목표 지점을 할당하여 움직임 인코딩한다. MODIA는 인스턴스화된 의사 결정 프로세스의 수가 선형적으로 증가함으로써 다루기 쉬운 상태를 유지한다. 이 방법은 교차로 시나리오에서 실제 자율주행차에서 테스트되었으며 교차로 시나리오에서 자율 차량 상호 작용을 성공적으로 해결하는 무지하고 순진한 기준선 알고리즘과 비교되었다.

5.2. 모션 계획(Motion Planning)

행동계층이 차선내에서의 순항, 차선변경, 우회전 등과 같은 현재의 전후관계(context)에서 수행되어야 할 운전동작을 결정할 때 선택한 동작이 낮은 레벨 피드백 제어기에서 추적될 수 있는 경로 또는 궤적으로 변환되어야 한다.

그 결과 경로 또는 궤적은 자동차가 동적으로 실행 가능해야 되고, 승객들이 편안하고 장착된 센스에 의해 검출된 장애물과의 충돌을 회피할 수 있어야 한다. 이와 같은 경로 또는 궤적을 구하는 과제는 모션 계획 모듈의 책임이 될 것이다.

모션 계획 모듈의 역할은 궤적을 생성해서 실제 물리적으로 자동차를 제어하는

피드백 제어 모듈로 보낸다. 이렇게 계획한 궤적은 보통 계획 궤적 지점 (trajectory point)의 시퀀스로 표현한다. 각 궤적 지점은 위치, 시간, 속도, 곡률 등과 같은 속성으로 구성된다. 자율 주행 자동차의 모션 계획 문제는 좀 더 일반적인 로봇틱스 모션 계획 문제의 특수한 경우로 볼 수 있다. 도로를 달리는 자율 주행차의 모션 계획 문제는, 자동차가 대부분 2차원 평면에서 기존 도로 그래프를 따라 이동하기 때문에, 어떻게 보면 로봇틱스에 적용되는 모션 계획보다 훨씬 쉽다. 제어 신호는 스로틀, 브레이크, 스티어링 휠에 대한 것만 있어서, 수행 가능한 궤적에 부드러움 및 곡률 제약 사항 같은 속성이 자연스럽게 표현되기 때문에 이보다 제약사항이 많은 (3차원 공간을 비행하는 드론에 대한 모션 계획 같은) 고차원 궤적을 계획하는 문제에 비해 최적화하기 훨씬 쉽다.

DARPA 어반 챌린지 이래로 자율 주행 자동차의 모션 계획 모듈을 점차 독립 모듈로 개발하는 추세다. [166, 167]에서는 시내 주행과 주차 같은 특정한 상황에 대한 모션 계획 문제를 해결하는 연구를 시도했으며 [162]와 같이 특수한 모션 계획 문제를 해결하는 연구도 진행됐다. [168, 169]는 다양한 측면에서 진행된 모션 계획에 대한 최신 연구 사례를 소개하고 있다. 이러한 연구 결과를 모두 참고해보면 자율 주행이라는 맥락에서 볼 때 모션 계획 모듈이 해결해야 할 문제가 점점 명확해지고 있다. 대부분의 모션 계획 모듈은 특정 시공간 제약 조건이 주어진 상황에서 시공간 궤적을 최적화하는 문제를 해결한다. 앞에서 설명했듯이 시공간 (spatial-temporal) 궤적은 여러 개의 궤적 지점으로 구성된다. 각 궤적 지점의 속성은 위치 시간, 속도, 가속도, 곡률 뿐만 아니라, 곡률 같은 속성에 대한 고차 미분 값을 비롯한 다양한 값으로 표현한다. 이러한 속성에 대한 비용에 따라 최적화 목표가 결정되기 때문에, 필수 제약 조건에 해당한다. 자동차 제어는 조화계 (harmonious system)가 아니기 때문에, 실제 자동차 궤적은 스플라인(spline) 궤적의 특성을 보인다. 따라서 모션 계획 문제는 2차원 평면에서 특정 공통 속성 및 제약 조건을 갖는 궤적에 대한 최적화 문제로 구성할 수 있다.

이렇게 구성한 최적화 문제에서 두 가지 핵심 요소는 최적화 객체(Optimization Object)와 제약 조건(Constraints)이다. 여기서 최적화 객체는 보통 각기 다른 후보 솔루션에 대한 비용 cost으로 표현하며, 최소 비용을 가진 객체를 찾는 것을 최적화의 목표로 삼는다. 비용을 계산하는 함수는 다음과 같은 두 가지 핵심 요소로 구성된다. 첫째, 비용 함수는 동작 결정 모듈에 직접 연결된 다운스트림 모듈이기 때문에 업스트림 동작 결정 모듈의 출력값을 반드시 따라야만 한다. 예를 들어 전방 자동차에 대해 일정 거리를 유지하고 추종하도록 독자 결정을 내렸다면 계획한 궤적은 독자 결정에서 명시한 지정 영역을 초과하지 않고 정확히 도달하도록 생성돼야 한다. 또한 계획한 궤적에서 충돌이 발생하지 않아야 하는데 이를 위해 궤적을 계산하는 동안 모든 물리 객체와 최소 거리를 유지해야 한다. 둘째 시내



도로에서 자율주행하는 데 초점을 맞추기 때문에 계획한 궤적은 도로 모양과 일관성을 유지해야 하는데, 자율 주행 자동차가 도로를 자연스럽게 따라서 이동해야 한다는 요구사항으로 일반적으로 해석한다. 이러한 사항은 모두 비용 함수의 설계에 반영된다. 비용 함수의 설계는 업스트림 동작 결정 모듈의 출력 값을 준수하며 경로 계획에서 제시하는 방향을 따르는 데 중점을 두는 반면, 모션 계획 최적화 문제는 다운스트림 피드백 제어 모듈이 자동차를 편안하게 움직이는 데 중점을 둔다. 예를 들어 곡률 및 곡률의 2차 미분 값은 스티어링 휠 제어에 대한 제약 조건을 표현한다. 마찬가지로 스로틀로 가속을 제어할 때 가속도의 변화율도 제한한다.

모션 계획을 위한 방법은 크게 그래프 검색 기반, 샘플링 기반, 보간 곡선 기반 및 수치 최적화 기반 방법의 4가지로 분류할 수 있다[170, 171].

5.2.1. 그래프 검색 기반 기법

궤적 계획을 위한 그래프 검색 기반 기술은 시간의 경과에 따른 차량 상태의 진화를 명시하기 위해 경로 계획을 위한 기법을 확장한다. 자율주행차에 대한 가장 일반적인 그래프 검색 기반 궤적 계획 기법은 상태 격자, 탄성 밴드(EB) 및 A*이다.

상태 격자는 정점이 상태를 나타내고 가장자리가 로봇의 운동학적 제약 조건을 만족하는 상태를 연결하는 경로를 나타내는 검색 그래프이다[172]. 꼭짓점은 규칙적인 방식으로 배치되며, 비록 엄격하게 변환되고 회전되지만, 각 꼭짓점에서 목표까지의 실현 가능한 경로를 구축하는 데 사용될 수 있다. 이러한 방식으로, 목표에 대한 궤적은 그래프에서 일련의 가장자리로 표현될 가능성이 있다. 상태 격자는 위치, 속도 및 가속도와 같은 여러 차원을 처리할 수 있으며 동적 환경에 적합하다. 그러나 그래프에서 가능한 모든 솔루션을 평가하기 때문에 계산 비용이 높다[170].

McNaughton 등은 궤도 계획을 위한 등각 시공간 상태 격자를 제안한다. 그들은 중심선 경로를 중심으로 상태 격자를 구성한다[173]. 그들은 도로의 노드를 중심선에서 측면 오프셋으로 정의하고 최적화 알고리즘을 사용하여 노드 사이의 에지를 계산한다. 이 최적화 알고리즘은 노드 쌍을 연결하는 가장자리를 정의하는 다항식 함수의 매개 변수를 찾는다. 그들은 각 노드에 포즈, 가속도 프로파일, 시간 및 속도 범위를 포함하는 상태 벡터를 할당한다. 가속 프로파일은 시간과 속도 간격의 미세한 이산화보다 적은 비용으로 궤적 다양성을 증가시킨다. 또한, 시간과 속도의 범위는 그래프 구성 단계 대신 그래프 검색 단계에 시간과 속도를 할당할 수 있으므로 계산 비용을 절감한다. Xu 등은 McNaughton 등[173]이 제안한 상태



격자 궤적 계획에서 파생된 결과 궤적에 적용되는 반복 최적화를 제안하여 계획 시간을 단축하고 궤적 품질을 향상시킨다[174]. Gu 등은 McNaughton 등[173]이 제안한 상태 격자 궤적 계획과 전술적 추론을 융합한 계획 방법을 제안한다[175]. 후보 궤적의 세트는 다른 기동이 추출되는 상태 격자에서 샘플링된다. 최종 궤적은 기동(예: 차선 유지 또는 차선 변경)을 선택하고 선택한 기동과 관련된 후보 궤적을 선택하여 얻는다. Li 등은 큐빅 다항식 곡선을 사용하여 경로를 따라 후보 경로를 생성함으로써 상태 격자를 구축한다[176]. 속도 프로파일은 또한 생성된 경로의 포즈에 할당되도록 계산된다. 결과 궤적은 비용 함수에 의해 평가되고 최적의 궤적이 선택된다.

또 다른 그래프 검색 기반 기술은 탄성 대역으로, 처음에는 경로 계획에서 사용되었고 나중에는 궤도에서 사용되었다. 탄성 밴드는 탄성 정점과 모서리가 있는 검색 그래프이다. 탄성 정점은 인접한 공간 정점을 연결하는 인-엣지와 아웃-엣지로 공간 정점을 증가시킴으로써 정의된다. 경로는 두 가지 힘, 즉 외부 장애물에 의해 생성된 반발력과 밴드 느슨함을 제거하기 위해 인접 지점에 의해 생성된 수축력의 균형을 맞추는 최적화 알고리즘에 의해 발견된다. 이 기술은 연속성과 안정성을 보여주며, 비결정적 런타임은 있지만 충돌이 없는 초기 경로가 필요하다.

Gu 등은 분리된 시공간 궤적 계획 방법을 제안한다. 궤도 계획은 세 단계로 나뉜다[177]. 첫 번째 단계에서, 충돌 없는 실현 가능한 경로는 순수 추적 제어기와 운동학적 자동차 모델을 사용하여 도로와 장애물 제약 조건을 고려하여 탄성 대역에서 추출한다. 두 번째 단계에서 속도 프로파일은 속도 제한, 장애물 근접, 횡방향 가속도 및 종방향 가속도 등 몇 가지 제약 조건하에서 제안된다. 마지막으로, 경로와 속도 프로파일이 주어지면, 궤적은 매개변수 경로 나선형을 사용하여 계산된다. 궤적은 향후 움직임을 시뮬레이션하여 모든 정적 및 움직이는 장애물에 대해 평가된다.

A* 알고리즘은 일반적으로 경로 계획에 사용된다. Fassbender 등은 궤적 계획을 위한 두 가지 새로운 A* 노드 확장 계획을 제안한다[178]. 첫 번째 계획은 수치 최적화를 사용하여 자동차의 현재 노드를 목표 노드에 직접 연결하는 궤적을 찾으려고 한다. 두 번째 체계는 순수 추적 컨트롤러를 사용하여 글로벌 기준 경로를 따라 차를 안내하는 짧은 에지(즉, 짧은 모션 프리미티브)를 생성한다.

5.2.2. 샘플링 기반 기법들

샘플링 기반 기술은 자동차의 현재 상태와 다음 목표 상태 사이의 연관성을 찾기 위해 상태 공간을 무작위로 샘플링한다. 자율주행차의 궤도 계획에 가장 많이 사용되는 샘플링 기반 기술은 RRT(Rapid-exploring Random Tree)이다.



궤도 생성을 위한 RRT 방법은 상태 공간의 무작위 샘플을 사용하여 자동차의 현재 상태에서 검색 트리를 점진적으로 구축한다[142]. 각 랜덤 상태에 대해, 제어 명령은 가능한 한 랜덤 상태에 가까운 새 상태를 만들기 위해 트리의 가장 가까운 정점에 적용됩니다. 트리의 각 꼭짓점은 상태를 나타내며 각 방향 에지는 상태를 확장하기 위해 적용된 명령을 나타냅니다. 후보 궤적은 다양한 기준에 따라 평가된다. 그림 8은 현재 자동차의 상태에서 무작위 상태로 RRT 방식으로 생성된 궤적의 예를 보여준다. RRT 방법은 고차원 공간에 대한 계산 비용이 낮으며 해결 방법이 있고 충분한 시간이 주어지면 항상 해결 방법을 찾는다. 그러나 결과는 연속적이지 않고 극단적이다[170].

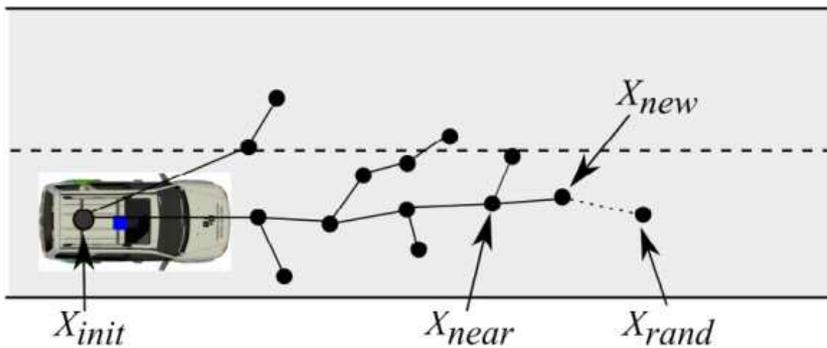


그림 5.1. RRT를 이용한 차량 궤적 생성의 예(출처: [170])

Radaelli 등은 자율주행차 IARA의 궤도 계획을 위한 RRT 방법을 제안한다 [179]. 그들은 경로를 향해 무작위 상태의 위치를 편향하기 위해 표준 RRT에 새로운 변형을 제시하고, 상태를 확장하기 위해 가장 유망한 제어 명령을 선택하고, 유망하지 않은 상태를 폐기하며, 최상의 궤적을 구성하는 상태 집합을 선택한다. 또한 이전 계획 주기에 구축된 궤적의 일부를 재사용한다. Du 등은 도로에서의 운전자의 시각적 검색 행동을 이용하여 RRT의 상태 샘플링을 안내하는 RRT 방법을 제안한다[180]. 운전자들은 곡선 도로를 운전하기 위해 “근접 지점”과 “먼 지점”을 모두 사용한다. 그들은 곡선 도로에서 나타난 운전자의 시각적 검색 행동의 이러한 특성을 이용하여 RRT 방법을 안내한다. 또한, 그들은 부드럽고 연속적이며 실현 가능한 궤적을 생성하기 위해 B-스플라인을 기반으로 한 후 처리를 채택한다.

5.2.3. 보간 곡선 기반 기법들

보간 곡선 기반 기술은 이전에 알려진 일련의 포즈(방향과 위치)를 보간하고 다른 매개 변수 중에서 자동차의 운동학적 및 동적 제약, 편안함, 장애물을 고려하는 더 부드러운 궤적을 구축한다. 자율주행차의 궤도 계획을 위한 가장 일반적인

보간 곡선 기반 기법은 클로토이드 곡선이다[170].

직교 곡선은 직선 가변 곡률로 궤적을 정의할 수 있으므로 직선 세그먼트에서 곡선 세그먼트로의 전환이 부드럽다. 그러나 클로소이드 곡선은 이를 정의하는 적분 때문에 계산 비용이 높고 전역 웨이포인트에 따라 달라진다[170]. Alia 등은 궤도 계획을 위해 클로소이드 텐타클 기법을 사용한다[181]. 텐타클은 차의 무게 중심에서 시작하여 전체의 형태를 취하면서 다른 속도와 다른 초기 조향 각도에 대해 계산된다. 텐타클은 텐타클의 기하학적 매개 변수에서 조향 각도 명령을 도출하고 점유 그리드 맵에서 충돌 검사를 실행하는 경로 추적 컨트롤러를 사용하여 탐색 가능 또는 탐색 불가능으로 분류된다. 항해할 수 있는 텐타클 중에서 몇 가지 기준에 따라 최고의 텐타클 선택한다. Mouhagir 등도 궤도 계획을 위해 클로소이드 텐타클 기법을 사용한다[182, 183]. 그러나 그들은 최고의 텐타클을 선택하기 위해 MDP에서 영감을 받은 접근 방식을 사용한다.

5.2.4. 수치 최적화 기반 기술들

수치 최적화 기반 기술은 제한된 변수를 사용하여 함수를 최소화하거나 최대화한다. 자율주행차의 궤도 계획을 위한 가장 일반적인 수치 최적화 기반 기술은 기능 최적화와 모델 예측 방법이다.

함수 최적화 방법은 위치, 속도, 가속도 등 궤적의 제약을 고려하는 비용 함수를 최소화하여 궤적을 찾는다. 이러한 방법은 자동차의 운동학적, 동적 제약 및 환경의 제약 조건을 쉽게 고려할 수 있다. 그러나 최적화 프로세스가 각 모션 상태에서 발생하고 전역 경유지에 의존하기 때문에 계산 비용이 높다[170]. Ziegler 등은 자율 주행차 버스(Bertha)의 궤도 계획을 위해 기능 최적화 방법을 사용한다[65]. 그들은 비용 함수를 최소화하고 궤적의 제약을 준수하는 최적의 궤적을 찾는다. 비용 함수는 궤적이 지정된 속도로 주행 코리더의 중간(경로와 동일)을 따르도록 하고, 강한 가속도를 불이익시키고, 가속도의 급격한 변화를 감소시키며, 높은 요 레이트를 감소시키는 일련의 항으로 구성된다.

궤도 계획을 위한 모델 예측 방법은 미래 상태를 예측하는 모델을 사용하여 자동차의 현재 상태와 다음 목표 상태 사이에서 동적으로 실현 가능한 제어 명령을 생성한다[184]. 예를 들어, 역학이 미분 방정식으로 표현될 수 있는 상태 제약 조건을 충족하는 매개 변수화된 제어 명령을 생성하는 문제를 해결하는 데 사용될 수 있다. Ferguson 등은 자율주행차 Boss (2007 DARPA Urban Challenge에서 1위를 차지한 카네기 멜론 대학의 자동차)의 궤도 계획을 위해 모델 예측 방법을 사용한다[185]. 그들은 중심선 경로에서 파생된 목표 상태 집합에 대한 궤적을 생성한다. 각 궤적을 계산하기 위해, 그들은 궤적 끝점 오류가 허용 가능한 경계 내에



도달할 때까지 궤적 제어 매개 변수의 초기 근사치를 점진적으로 수정하는 최적화 알고리즘을 사용한다. 궤적 제어 매개 변수는 곡률 프로파일을 정의하는 스플라인 곡선의 궤적 길이와 세 개의 매듭점이다. 속도 프로파일은 현재 도로의 속도 한계, 최대 실현 가능 속도 및 목표 상태 속도를 포함한 몇 가지 요소를 기반으로 각 궤도에 대해 생성된다. 장애물과의 근접성, 중심선 경로까지의 거리, 평활성, 끝점 오류 및 속도 오류에 따라 최적의 궤도가 선택된다.

Li 등은 경로를 따라 목표 상태를 샘플링하는 상태 샘플링 기반 궤적 계획 체계를 사용한다[186]. 모델 예측 경로 계획 방법은 자동차의 현재 상태를 샘플링된 목표 상태와 연결하는 경로를 생성하기 위해 적용된다. 속도 프로파일은 생성된 경로를 따라 각 상태에 속도를 할당하는 데 사용된다. 안전성과 편안함을 고려한 비용 함수를 사용하여 최상의 궤적을 선택한다.

Cardoso 등은 자율주행차 IARA의 궤도 계획을 위해 모델 예측 방법을 사용하였다[187]. 궤적을 계산하기 위해 목표 상태까지의 거리, 경로까지의 거리, 장애물과의 근접성을 최소화하는 궤적 제어 매개 변수를 찾는 최적화 알고리즘을 사용한다.

5.3. 피드백 제어

모션 계획 시스템으로부터 기준 경로 또는 궤적을 실행하기 위해서는 계획된 움직임을 수행하고 트래킹 오차를 교정하기 위한 적절한 액추에이터 입력을 선택하는데 피드백 제어가 사용된다. 계획된 움직임의 실행 동안에 발생한 트래킹 오차는 부분적으로 자동차 모델의 부정확함에도 기인한다.

그림 5.2에서 보여주고 있는 것처럼 피드백 제어기의 역할은 모델링 오차나 다른 형태의 불확실성이 존재해도 기준 경로나 궤적을 안정화시키는 것이다. 특히 제어기는 특정 오차를 생성하는 참조에 대한 시스템 출력을 비교한다. 측정된 오차를 근거로 제어기는 새로운 시스템 입력을 생성한다.

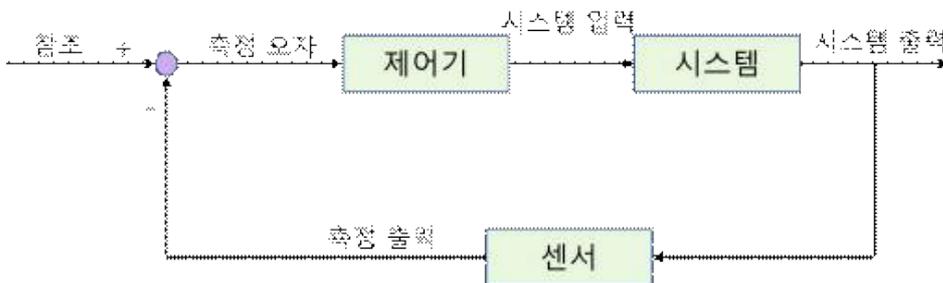


그림 5.2. 피드백 제어

자율 주행차 제어는 사전 정의된 특정 궤적을 따르며 지속적인 피드백을 통해



실제 자세와 사전 정의된 궤적상의 자세 간 오차를 추적한다. [142]에서 저자들은 자율 주행차 피드백 제어에 관한 많은 연구를 소개하는데 그중에서 [188, 189]는 기존의 피드백 제어 시스템에 장애물 회피 및 경로 최적화를 추가로 도입했다. 앞에서 제안한 자율 주행차 계획 및 제어의 시스템 아키텍처를 통해, 피드백 제어 모듈도 기존의 자동차 자세 피드백 제어 기법을 최대한 활용할 수 있다.

5.3.1. 직접 하드웨어 작동 제어 기법

직접 하드웨어 작동 제어 방법은 모션 계획 모듈에 의해 계산된 궤적으로부터 자동차의 조향, 스로틀 및 브레이크 액추에이터에 대한 힘 입력을 직접 계산한다.

그리고 자율 주행차의 속도와 스티어링 휠 각도에 영향을 미치는 모델에 의해 주로 야기되는 부정확성을 완화한다.

자율주행차에 대한 가장 일반적인 직접 하드웨어 작동 제어 방법 중 하나는 피드백 제어이다. Funke 등은 피드백 제어 방법을 사용하여 Audi TTS의 컨트롤러 서브시스템을 구현하였다[190]. 아우디 TTS의 하드웨어 구조는 200Hz에서 실시간 제어가 가능하도록 설계되었다. 이 고속 하드웨어는 그들의 컨트롤러 서브시스템이 최대 160km/h의 속도로 차를 운전할 수 있게 해주었다. Ziegler 등은 버사(Bertha)의 제어 모듈에 피드백 제어 방법을 사용하였다[65]. 버사의 제어 모듈은 103km 길이의 버사-벤츠-메모리얼-루트를 통해 최대 100km/h의 속도로 차량을 주행할 수 있었다. Li 등은 도요타 랜드크루저에서도 동일한 제어 방식을 채택하여 최대 25km/h의 속도로 주행할 수 있었다[186].

5.3.2. PID 제어

자율주행차에 널리 사용되는 또 다른 하드웨어 작동 방법은 PID 제어 방법이다. 여기에는 원하는 하드웨어 입력과 하드웨어 출력이 원하는 하드웨어 입력에서 얼마나 멀리 떨어져 있는지를 설명하는 오류 측정이 포함된다[191].

그림 5.3에서, PID 제어기는 원하는 설정점과 관측된 공정변수간의 차이에 따른 오차값을 연속적으로 계산하고, 비례, 적분, 미분항을 기반으로 한 교정을 적용한다.

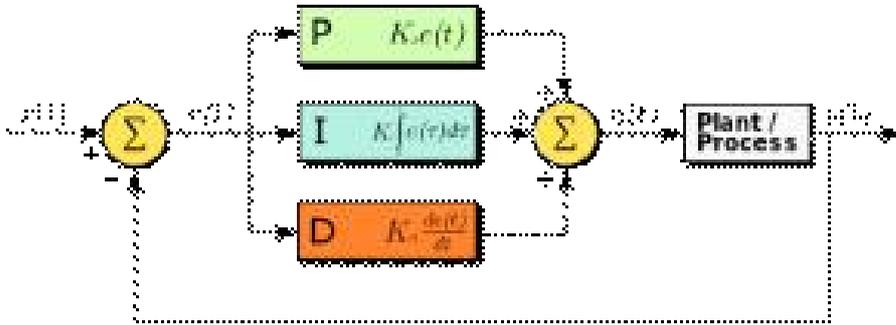


그림 5.3. PID 기반 피드백 제어 시스템[192]

PID 제어의 공통적인 예는 자동차의 순항제어인데 오르막에서 일정한 엔진 파워만을 적용하면 속도가 낮아질 것이다. 제어기의 PID 알고리즘은 엔진의 파워 출력을 증가시켜서 최소의 지연과 오버슈터(overshoot)로 원하는 속도가 되도록 측정 속도를 복구시킨다.

PID 제어기의 두드러진 특징은 정확한 최적제어를 적용하기 위해 제어 출력에 영향을 미치는 비례, 적분, 미분의 세 가지 제어항을 사용할 수 있는 능력이다. 제어기는 제어 밸브의 개방과 같은 제어변수를 제어항의 가중치 합으로 결정되는 새로운 값으로 조정함으로써 시간에 걸친 오차를 최소화되도록 한다.

피드백 제어 기법과 유사하게, PID 제어 방법은 시간이 지남에 따라 누적된 과거 오류 외에도 현재 오류를 처리할 수 있다[193]. Zhao 등은 자율 주행차 Intelligent Pioneer의 컨트롤러 서브시스템을 구현하기 위해 일반화 최소 분산 기법을 기반으로 한 적응형 PID 방법을 사용하였다[194]. 인텔리전트 파이오너의 제어 모듈은 최대 시속 60km로 차를 운전할 수 있었다.

Levinson 등은 로봇 자동차 Junior의 컨트롤러 서브시스템에 모델 예측 제어(MPC) 방법과 피드포워드 PID 제어 기법을 혼합하였다(Stanford University의 자동차는 2007 DARPA Urban Challenge에서 2위를 차지했다)[195]. 제어 모듈은 입력을 수신하고 스톱, 브레이크 및 스티어링에서 작동하기 위한 노력을 계산한다.

5.3.3. 모델 예측 제어(Model Predictive Control)

모델 예측 제어 (MPC)는 일련의 제한조건을 만족하면서 공정을 제어하는데 사용되는 제어의 발전된 방법이다[196, 197]. 그림 5.4에서 보여주고 있는 바와 같이, MPC는 공정의 동적 모델 즉 대체로 시스템 식별에 의해 얻어진 선형인 경험적 모델에 의지한다. MPC의 주요 이점은 미래의 타임슬롯을 고려하면서 현 타임슬롯이 최적화가 되게 하는 것이다. 특히 MPC는 미래의 사건을 예상하는 능력을 갖고 이에 따라 제어 동작을 취할 수 있는데 반해 PID 제어기는 이러한 예측능력을 갖지 못한다.



자율 주행차에 대한 MPC 활용의 상세한 예는 문헌 [198]에서 찾을 수 있는데 다음과 같은 구성요소를 포함하고 있다.

- 자동차 제어 인터페이스: 자동차는 100ms마다 조향 및 스로틀(throttle: 카브레이터 등의 조절판)을 조정한다.
- 비용함수: 하이레벨상에서 비용함수는 목표 궤적 포인트와 실제 자동차 궤적간의 차이를 나타낸다. 구체적인 비용함수는 교차트랙 오차, 헤딩오차, 속도비용, 조향비용, 가속도비용, 조향 변화율 및 가속도 변화율 등의 가중치 합이다.
- 제한조건: 자동차 휠은 25° 이상 조향할 수 없다.

이 예에서 MPC는 제한조건을 만족하면서 비용함수를 최소화시키는데 활용될 수 있다. 각 주기마다 (100m) 자동차의 현재상태를 결정하기 위해 자동차의 위치 (x, y), 속도(v), 방향(ψ), 조향각(δ) 및 가속도(a) 등과 같은 센서 데이터를 읽어 들인다. 그러면 MPC는 읽어 들인 센서 데이터를 기반으로 짧은 시간 (가령 1초)내에 가능한 동작을 생성한다.

예로 MPC에 의해 생성된 최적 계획이 휠을 시계방향으로 20° 조향시키고, 매 100ms 마다 1° 씩 조향을 줄여나감과 동시에 1초 주기의 마지막에서 비용함수의 최소화를 기대하는 것으로 가정해보자. MPC는 휠을 20° 움직이게 하기 위한 첫 번째 동작을 수행한다. 다음 사이클(다음 100ms)에서 MPC는 다시 센서 입력을 읽어 들인다. 새로운 센싱으로 남은 동작을 수행하는 대신, MPC는 다음 최적 동작을 다시 계산하는 과정을 반복한다.

MPC의 아름다움은 단순히 한 스텝 동작을 생성하는 대신 1초 또는 10 스텝과 같은 보다 먼 미래계획을 고려하면서 다음 행동을 반복적으로 생성하는데 있다. 결과적으로 PID와 달리, MPC는 그리디(greedy)방법에서 짧은 기간의 이득에 덜 취약하게 되고 이로 인해 부드러운 궤적을 생성할 수 있다.

Levinson 등은 MPC와 PID를 혼합하여 주니어를 최대 56km/h까지 운전할 수 있도록 했다[195]. Guidolini 등은 자율주행차 IARA의 스티어링 휠 하드웨어 지연을 해결하기 위해 신경 기반 모델 예측 제어(N-MPC) 방법을 제안하였다[193]. 그들은 MPC 방법을 사용하여 현재 궤도에 따라 자동차를 적시에 이동시킬 수 있는 노력을 예상함으로써 조향 하드웨어 지연의 영향을 줄였다. 그들은 신경망을 사용하여 IARA의 조향 하드웨어를 모델링하고 N-MPC 조향 제어 방법에 신경 기반 조향 모델을 사용했다. 제안된 솔루션은 PID 제어 방법을 기반으로 표준 솔루션과 비교되었다. PID 제어 모듈은 최대 25km/h의 속도에서 잘 작동했다. 그러나 이 속도 이상이 되면 IARA의 스티어링 하드웨어의 지연이 너무 커서 제대로 작동할 수 없었다. N-MPC 모듈은 IARA의 조향 하드웨어 지연의 영향을 줄였고, 이는 IARA의 자율적 작동이 최대 37km/h까지 가능하게 하여 이전의 최대 안정 속도보다 48%



증가하였다.

5.3.4. 경로 추적 기법

경로 추적 방법은 주로 자동차의 모션 모델에 의해 야기되는 부정확성을 줄이기 위해 모션 계획 모듈에 의해 계산된 모션 계획의 실행을 안정화(즉, 오류 감소)시킨다. 장애물을 다루지는 않지만 단순화된 궤적 계획 기법으로 간주할 수 있다. 순수 추적 방법은 구현이 간단하기 때문에 경로 추적을 위한 자율 주행 차량에 널리 사용된다[171]. 그림 5.4에 도시된 바와 같이, 현재 경로로부터 어느 정도 미리 내다보는 거리에서 경로의 지점을 찾고, 원호가 후방 차축의 중심과 경로의 지점을 연결하도록 전륜을 회전시키는 것으로 구성된다[171]. 순수 추적 방법을 개선하기 위해 제안된 변형 세트가 있다. Samson은 모션 계획의 실행을 안정화하기 위해 후륜 위치를 하드웨어 출력으로 사용할 것을 제안했다[199]. Thrun 등은 전륜 위치를 규제 변수로 하는 자율주행차 Stanley의 제어 접근법을 제시했다[37]. 스탠리의 제어 모듈은 최대 60km/h의 속도로 차를 운전할 수 있었다.

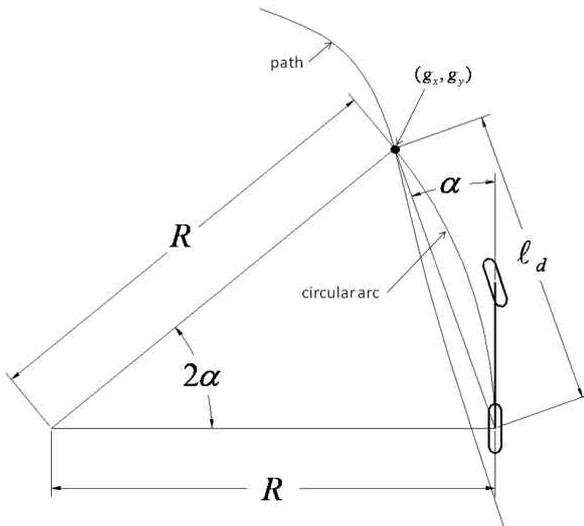


그림 5.4. 순수 추적 기하학(출처: [200])

모델 예측 제어(MPC) 방법은 자율주행차에 널리 사용된다. 미래 예측 지평선에 걸쳐 출력을 시뮬레이션하고 최적화하기 위해 자동차의 모션 모델을 사용하여 원하는 하드웨어 출력으로 이어질 제어 명령 입력을 선택하는 것으로 구성된다[193]. Koga 등은 소형 전기 자동차의 경로 추적을 위해 MPC 방법을 사용한다[201]. 그들은 자동차의 움직임을 예측하기 위해 표준 자전거 모델을 사용한다. 전기차의 컨트롤러 서브시스템은 최대 20km/h의 속도로 차를 운전할 수 있었다. Kritayakirana 등은 아우디 TTS의 제어 접근법을 제시하는데, 이 접근 방식은 자동차의 동적 모션 모델을 사용하여 취급 한계에서 경로 추적을 가능하게 한다. 자



동차의 동적 운동 모델은 타이어 미끄러짐으로 인한 경로 계획으로부터의 편차를 고려한다. 실험 결과 아우디 TTS는 최대 160km/h의 속도를 낼 수 있었다 [202-204].

5.4. 결론

4장과 5장에서는 경로 계획 모듈, 동작 결정 모듈, 트래픽 예측 모듈, 모션 계획 모듈, 피드백 제어 모듈로 구성된 자율주행 자동차 계획 및 제어의 일반적인 아키텍처에 대해 설명했다. 이러한 모듈에 대해 학계와 업계에서 제안한 방식이 다양하게 나와 있다. 각 제안 방법마다 나름대로 이론적 또는 실용적으로 입증된 장점이 있다. 실제로 자율 주행에서 해결할 난제는 개별 모듈 단위 문제가 아니라, 자율 주행 자동차 계획 및 제어 문제 전체를 계층으로 분할해서 각각을 일관성 있게 해결하는 방법에 대한 것이다. 이러한 관점에서 볼 때 여기서는 현존하는 솔루션을 모두 나열하는 방식으로 살펴보지 않았다. 그보다는 각 모듈 사이의 일관성 있는 연동을 통해 해결하는 관점에서 살펴봤다. 각 모듈의 문제와 역할 범위를 명확히 정의하고, 각 모듈이 주어진 문제를 데이터 흐름에 따라 추상적인 수준에서 구체화하는 방법을 설명하는 데 초점을 뒀다. 각 모듈 또는 각 계층마다 문제를 정의하고 현재 업계에서 해결하는 방법을 소개했다.



6. V2X 통신

6.1. 개요

자율 주행 시스템에 대한 스트레스를 완화하기 위한 효과적인 방법 중에서 하나는 차량 대 모든 사물 V2X(Vehicle to everything) 기술이다. V2X는 지능형 교통 시스템에서 차량과 다른 참여자(이동식 또는 정지식) 간의 외부통신을 지칭하는 일반적인 용어다. ETSI(European Telecommunications Standards Institute)는 V2X 통신을 네 가지 유형으로 구별하는데, 차량 대 차량(V2V), 차량 대 인프라(V2I), 차량 대 네트워크(V2N), 차량 대 보행자(V2P)이다[205]. V2V는 서로 근접한 차량 간의 정보 교환에 초점을 맞춘다. V2I는 도로변 장치(RSU) 라고도 하는 지능형 도로 인프라와 차량 간의 직접 통신을 나타낸다. V2N은 차량과 인터넷간의 통신을 의미한다. V2P는 보행자, 자전거 타는 사람 등과 같은 인간 교통 참여자와의 차량 통신을 의미한다.

현재 대부분의 연구는 V2V와 V2I에 초점을 맞추고 있다. 기존 자율주행 시스템은 차량 내부에 고가의 센서와 첨단 컴퓨팅 장비가 필요하지만, V2X는 도로 인프라 투자와 같은 다른 접근 방식을 취함으로써 차량의 컴퓨팅 및 감지 비용을 경감시킨다.

V2X의 궁극적인 목표는 차량과 다른 교통 참여자들이 사고를 피할 수 있도록 도와서 교통 안전을 개선하는 것이다. V2X 통신에서 얻은 정보는 처리돼 인간 운전자에게 경고 형태로 줄 수 있으며, 자동화 차량에서 일부 안전 메커니즘을 작동시킬 수 있다. V2X는 자율 주행 자동차가 가시거리와 내부 정보(예: 공사 또는 사고로 인해 차선 방향이 일시적으로 변경되거나 갑작스런 기상 악화로 인해 인식 기능이 일시적으로 제한되는 경우)를 넘어서는 중요한 상황을 인지할 수 있도록 한다.

V2X의 또 다른 중요한 목표는 특히 트래픽과 에너지 효율의 맥락에서 효율성을 높이는 것이다. 지역적인 V2X 정보의 도움을 받아, 교통 혼잡을 줄이거나 전체 교통 흐름을 개선하려면 속도를 조정하도록 차량에 알릴 수 있다. 또한 이 동적 정보를 사용해 에너지 소비를 최적화하거나 주행과 관련된 환경 비용을 절감할 수 있다

이 장에서는 V2X 기술발전에 대해 논의하고, 자율주행을 위한 V2X의 몇 가지 사용사례, 즉 호송주행, 협동차선변경, 협동교차로관리, 협동감지 등을 제시한다.

6.2. V2X 표준

V2X를 최대한 활용하려면 가능한 한 다양하고 많은 업체가 적극적으로 참여해야 하므로 표준화는 다양한 제조업체의 기기(또는 스테이션)간 상호 운용성을 보장하려면 반드시 필요하다. 아이러니하게도 V2X는 IEEE 802.11p 기반 표준과 최근의 3GPP C-V2X(Cellular Vehicle-to-Everything) 표준이라는 두 개의 독립적인 표준으로 분리됐다. IEEE 802.11p 표준은 차량 네트워크 통신에 WiFi를 사용할 수 있게 하는 IEEE 802.11 WiFi 표준의 개정판이다. 그림 6.1과 같이 DSRC(Dedicated Short Range Communication)와 유럽 ETSI ITS-G5 V2X(그림 6.2) 표준 모두 프로토콜 스택의 물리(PHY) 및 매체 접근 제어(MAC) 계층에 IEEE 802.11p를 사용한다 [206]. C-V2X 표준은 2016년 설립된 자동차·기술·통신 업계 기업 컨소시엄인 5GAA(5G Automotive Association)가 개발했다. C-V2X 표준은 V2X 통신을 위해 셀룰러(LTE-4G 및 5G)를 사용하므로 물리 계층이 IEEE 802.11p와 완전히 호환되지는 않지만, DSRC와 ITS-G5 표준의 상위 계층 프로토콜과 서비스를 재사용한다.

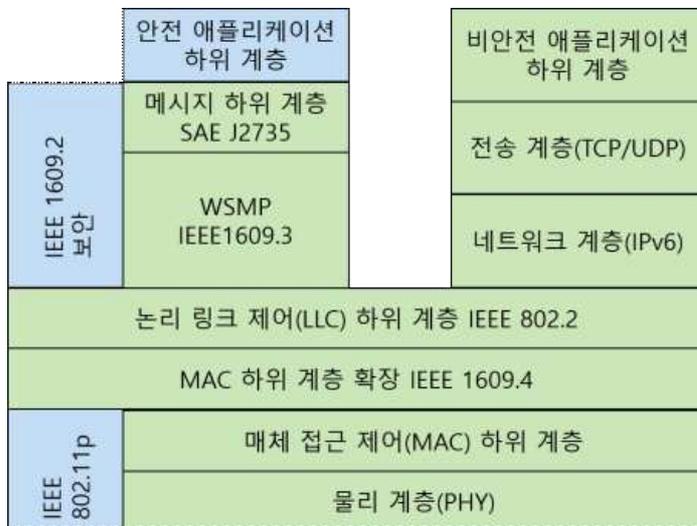


그림 6.1. DSRC 프로토콜 스택[207]

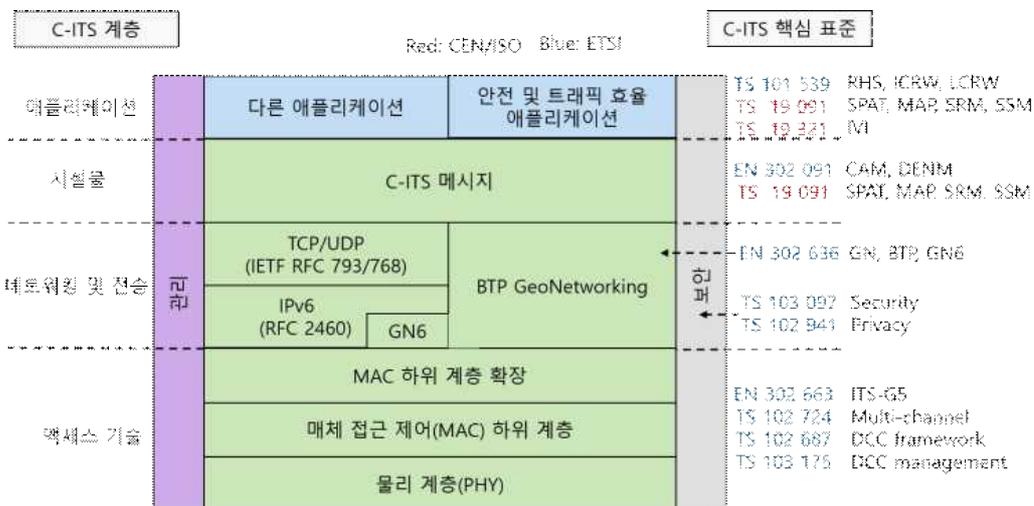


그림 6.2. ITS-G5 프로토콜 스택[208]

모든 V2X 표준에는 안전 중요 애플리케이션과 비안전 중요 애플리케이션을 위한 서로 다른 프로토콜 스택이 있다. 비안전 중요 애플리케이션은 일반적으로 전송 계층 및 네트워크 계층 프로토콜에 각각 TCP/UDP와 IPv6 사용한다. 안전 중요 애플리케이션은 전송 및 네트워크 계층의 프로토콜이 표준마다 다르다. DSRC는 IEEE 1609.3 WSMP(WAVE Short Message Protocol)[209]를 사용하는 반면 ITS-G5는 BTP(Basic Transport Protocol)[210]와 GeoNetworking 프로토콜[211]을 사용한다. 유럽 표준을 기반으로 V2I 참가자 간의 안전 중요 메시지는 CAM(Cooperative Awareness Message)과 DENM(Decentralized Environment Notification Message)으로 교환되며, CAM은 10Hz와 1Hz 사이에서 주기적으로 전송되고 V2X 네트워크의 다른 참가자에게 상태 정보(예: 병행, 속도, 차선 위치)를 주기적으로 제공한다. DENM은 이벤트 기반 메시지로서 트래픽 걸림 감지 등 트리거 조건이 충족될 때마다 트리거되며, 종료 조건(예: 트래픽 걸림 종료)에 도달할 때까지 반복적으로 전송된다. DSRC 기반 V2X 네트워크는 SAE J2735 표준에 정의된 안전 메시지 세트를 사용한다. 예를 들어, 신호교차로의 상태 통신을 위한 SPAT(Signal Phase and Timing) 메시지와 도로 표지 유형 및 권장 속도 같은 관련 정보를 교환하기 위한 TIM(Traveler Information Message)이 있다[212].

6.3. V2X 기술의 발전

V2X 기술의 개발에 있어서, 많은 연구자들이 V2X 통신 프로토콜의 특정 난제에 대한 해결책을 제시했다. 차량간 위험 경고 시스템(IVHW :Inter-Vehicle Hazard Warning system)은 통신을 기반으로 차량 안전성을 향상시키는 아이디어를 취하기 위해 가장 먼저 연구된 시스템 중 하나이다. IVHW는 869MHz의 주파수 대역에서 경고 메시지가 방송 메시지로 전송되는 통신 시스템이다[213].

IVHW는 현지 의사결정 전략을 취한다. 차량이 메시지를 수신한 후 관련 점검을 수행하여 경고 메시지가 관련 되는지, 그리고 운전자에게 보여야 하는지 여부를 결정한다. 연구노력의 대부분은 관련 체크 알고리즘의 설계에 있었다. 그러나 IVHW가 메시지를 공유하기 위한 방송 메커니즘을 취함에 따라 대역폭과 컴퓨팅 자원 모두에서 엄청난 낭비가 발생할 수 있다.

IVHW의 브로드캐스트 메시지와 비교하여 ad-hoc 네트워킹은 멀티홉 차량 간 통신을 지원하는 더 나은 솔루션이 될 수 있다[214]. FleetNet은 차량 통신의 개념을 용한 또 다른 연구 프로젝트로서[215], ad-hoc 네트워킹을 기반으로 한다. 또한 FleetNet 프로젝트는 일부 인터넷 프로토콜 기반 애플리케이션을 위한 통신 플랫폼



품을 제공한다. FeetNet은 IEEE 802.11 무선 LAN(WLAN) 시스템을 기반으로 구현된다[216]. V2V 통신의 경우, 두 대의 차량이 무선으로 직접 연결되지 않은 경우, 이들을 위해 메시지를 전달하기 위해 일부 다른 차량이 필요할 것이다. 라우팅과 포워딩 프로토콜을 설계하는 것은 중요한 과제가 될 수 있다. 적응성과 확장성에 대한 요구사항을 충족하기 위해, FleetNet은 위치 기반 전달 메커니즘을 제안했다. 이 아이디어는 차량의 지리적 위치를 기준으로 메시지를 전달할 다음 홉을 선택하는 것이다.

CarTALK 2000은 협력적 운전자 지원 애플리케이션을 지원하기 위해 ad-hoc 통신망을 적용하는 데 초점을 맞춘 프로젝트이다[217]. 차량 네트워크 토폴로지는 동적이며 차량 수가 자주 변화하고 있으므로 V2V 통신에서 ad-hoc 기반 라우팅에 큰 어려움이 있을 수 있다[218]. 이 문제를 해결하기 위해 CarTALK 2000에서는 기본 도로 토폴로지와 같은 일부 공간 정보를 고려한 공간 인식 라우팅 알고리즘이 제안되었다. FleetNet과 비교했을 때, CarTALK 2000은 라우팅 알고리즘에 대한 추가 입력으로 공간 정보를 사용하기 때문에 더 나은 성능을 달성한다. CarTALK 2000과 FleetNet의 또 다른 유사점은 둘 다 WLAN 기술을 기반으로 한다는 것이다. AKTIV는 일부 운전 안전 애플리케이션에 셀룰러 시스템을 적용하는 또 다른 프로젝트다[219]. FleetNet 및 CarTALK 2000 프로젝트가 WLAN 기술을 기반으로 시스템을 구축한 이유 중 하나는 안전 관련 애플리케이션에 필요한 대기 시간이 500ms 미만이기 때문이다. 그러나 LTE(Long-Term Evolution) 통신 시스템이 크게 발전할 수 있다는 가정 하에 셀룰러 시스템이 희소 차량 네트워킹에 더 나은 선택이 될 수 있다.

한편, 일부 연구 프로젝트는 V2V 통신을 기반으로 한 경고 애플리케이션에 초점을 맞추고 있다. WILLWARN(Wireless Local Risk Warn)은 차량 내 데이터에 기반한 위험 감지 접근 방식을 제안했다. 경고 메시지에는 장애물, 도로 상태, 낮은 가시성 및 건설 현장이 포함된다[220]. WILLWARN은 V2X 기술 자체에 초점을 맞춘 다른 프로젝트와 달리 위험 지점과 같은 특정 시나리오에서 V2X 기술을 활성화하는 데 초점을 맞추고 있다. 특정 위치에서 잠재적인 위험이 감지되었지만 경고 메시지를 공유하기 위해 V2X 통신 기술을 지원하는 통신 범위에 차량이 없다고 가정한다[230]. 경고 메시지를 공유하기 위해 WILLWARN V2V 통신을 통해 위험 지점에 접근하는 차량에 경고 메시지를 전송하는 분산 배포 알고리즘을 제안했다.

프로젝트 네트워크 온 휠 NoW(Network on Wheels)는 802.11 WLAN 및 ad-hoc 네트워킹에 기반한 차량 통신을 구축하기 위해 FleetNet의 아이디어를 채택한다[231]. NoW의 목표는 이동성과 인터넷 애플리케이션을 모두 지원하는 통신 플랫폼을 설정하는 것이다. 예를 들어 네트워크 계층과 응용 계층을 모두 고려한 하이



브리드 포워딩 방식이 개발된다. 또한, 일부 보안 및 확장성 문제는 NoW에서 논의된다.

인프라도 V2X 기술에서 매우 중요한 역할을 하므로 일부 연구는 인프라와의 협력을 기반으로 하는 안전 애플리케이션 구축에 중점을 둔다. SAFESPOT은 도로변 인프라를 사용하여 운전 안전을 개선하는 것을 목표로 하는 통합 프로젝트다 [232]. 차량용 센서와 인프라 센서의 정보를 결합하여 SAFESPOT은 위험한 상황을 감지하고 실시간으로 경고 메시지를 공유한다. 또한, 경고 예측을 밀리 초 수준에서 초 수준으로 개선하여 운전자가 준비하고 조치를 취하는데 더 많은 시간을 제공할 수 있다. SAFESPOT에서는 위험 및 사고 경고, 속도 경보, 도로 이탈 방지, 협력 교차로 충돌 방지, 지원 및 긴급 차량에 대한 안전 여유를 포함하여 다섯 가지 응용 프로그램에 대해 논의한다[233].

2007년에 자동차간 통신 컨소시엄 C2C-CC(Car-2-Car Communication Consortium)라는 비영리 조직이 V2X 기술의 표준을 만들기 위해 서로 다른 프로젝트의 모든 솔루션을 결합하기 위해 설립되었다. 2010년부터 V2X 기술에 대한 작업의 초점은 연구 주제에서 전체 ITS의 실제 환경 배포로 옮겨졌다. 가장 인기 있는 배포된 프로젝트 중 하나는 실제 대도시 분야에서 V2X 애플리케이션을 테스트하는 것을 목표로 하는 simTD이다[234]. simTD에서 모든 차량은 IEEE 802.11p를 기반으로 하는 전용 단거리 통신(DSRC) 기술을 통해 서로 연결할 수 있다. 한편 차량은 IEEE 802.11p를 사용하여 도로변 인프라와 통신 할 수도 있다. simTD의 시스템 아키텍처는 ITS 차량 스테이션, ITS 휴게소 및 ITS 중앙 스테이션의 세 부분으로 나눌 수 있다. simTD에서 테스트하기 위한 애플리케이션에는 교통 상황 모니터링, 교통흐름 정보 및 내비게이션, 교통 관리, 운전 지원, 지역 위험 경고 및 인터넷 기반 애플리케이션이 포함된다.

Cellular Vehicle-to-everything(C-V2X)은 낮은 지연 시간의 V2V 및 V2I 통신을 제공하는 통합 연결 플랫폼으로 설계되었다[235]. 두 가지 통신 모드로 구성됩니다. 첫 번째 모드는 차량, 인프라 및 보행자 간의 직접 통신 링크를 사용한다. 두 번째 모드는 차량이 인터넷에서 정보를 수신할 수 있도록 셀룰러 네트워크를 활용하는 워크 통신에 의존한다. CV2X는 차량의 통신 범위를 더욱 확장하고 차량 정보 전송을 위한 대용량 데이터를 지원한다.

6.4. V2X를 이용한 자율 주행 기술

6.4.1. V2I를 이용한 자율 주행 기술

6.4.1.1. 도로 공사 경고

도로 공사는 의심할 여지없이 자율 주행 자동차에게 가장 어려운 운전 시나리

오 중 하나이다. 이는 새로운 주행 환경에 대한 정확한 동적 해석이 적시에 필요하기 때문이다. 좀 더 구체화해보면, 차량 메모리에 저장된 정보만을 의존하기에는 차선 폐쇄, 주행 차선 형상 및 차선 방향의 변화, 불분명한 차선 경계 표시(과거 차선 표시와 현재 차선 표시가 종종 혼재되어 있음), 속도 제한의 변화, 새로운 교통 표지, 사람에게로의 근접 등과 같은 상황이 많이 발생한다. 따라서 전방의 도로 공사에 대한 자율 주행 자동차의 경고는 반드시 유용하며, 내부 시스템을 새로운 도로 조건에 맞게 조정하거나 도로 공사를 완전히 피할 수 있는 대체 경로를 계산해 대비 할 수 있다. 도로 공사 경고의 예는 그림 6.3에 나와 있다.



그림 6.3. V2I 도로 공사 경고의 예(출처: 광주교통정보센터)

6.4.1.2. 도로 위험 및 사고 경고

도로 위험이란 도로 표면에서 운전을 위험에 빠뜨릴 수 있는 어떤 것을 말한다. 도로의 잔해, 동물, 얼음이 될 수 있다. 도로 위험 및 사고에 대한 조기 경고는 DENM 메시지를 이용해 잠재적 및 후속 사고를 방지하거나 진행 중인 교통 혼잡을 줄이기 위해 대체 경로를 탐색하도록 차량에 경고할 수 있다. 그림 6.4에 설명된 것처럼 이러한 경고는 차량의 센서가 모든 도로 위험(예: 얼음)을 완벽하게 감지할 수 없기 때문에 SDV에게 위험을 자각시켜줄 수 있는 좋은 수단이다.





그림 6.4. 도로 위험 경고의 예(출처: 광주교통정보센터)

6.4.1.3. 신호등 단계 이벤트

신호등 단계에 대한 정보와 현재 단계의 잔여 시간은 안전성과 효율성을 향상시키는 데 도움이 된다. 카메라와 같은 차량 센서를 이용한 신호등 단계 인식은 모든 교통 상황이나 기상 조건에서 항상 안정적으로 작동하지 않을 수도 있는데, 신호등 단계 정보를 활용하면 차량의 인식 정확성을 보완할 수 있다. SAE J2735 표준[212]에 정의된 SPAT 및 MAP 메시지를 활용해 각 차선에 대한 신호등의 현재 단계, 현재 단계에 대한 잔여 시간, 교차로의 물리적/기하학적 구조를 V2I 네트워크를 통해 교환할 수 있다. 그림 6.5는 V2I 신호등 단계 이벤트의 예제 애플리케이션을 보여준다.



그림 6.5. 신호등 단계 이벤트의 예(출처: 광주교통정보센터)

6.4.2. V2V 통신을 이용한 자율 주행 기술

6.4.2.1. 교차로 이동 보조 경고

미국 도로 교통안전국(NHTSA National Highway Traffic Safety Administration)의 분석에 따르면, 2008년 미국에서 발생한 교통사고의 '40%가 교차로와 관련이 있었다[236]. 또한 해당 간행물은 교차로 관련 사고의 주요 원인으로 '가려진 시야'를 꼽았다. 시야와 물리적 센서의 한계로 인해 차량에 탑재된 센서만 사용하는 자율 주행 자동차가 이러한 통계를 개선할 수 있을지는 불분명하다. 교차로 이동 보조 경고는 그림 6.6과 같이 교차로 주변의 다른 교통 참여자들에게 상세한 이동 정보를 제공함으로써 차량의 인식 능력을 시야 이상으로 확장시키고 이는 사고를 예방하는 데 큰 도움이 된다.



그림 6.6. 교차로에서의 보행자 경고의 예(출처: 광주교통정보센터)

6.4.2.2. 잘못된 방향 주행 경고

잘못된 방향 주행은 차선의 진행 방향과 반대로 주행하는 상황(예: 다른 차량을 향한 직진 주행)을 말한다. 고속으로 이동하는 고속도로에서는 정지 거리가 증가하기 때문에 잘못된 방향 주행은 특히 위험하다. 또한 고속도로에서는 이러한 주행을 회피할 수 있는 조치를 취할 수 있는 범위가 줄어들 수 있다. 그림 6.7에서와 같이 잘못된 방향으로 주행하는 운전자의 존재를 적시에 경고하는 것은 다른 차량에 대응할 기회를 만들 줄 수 있으므로 사고를 예방하는 데 도움이 된다.

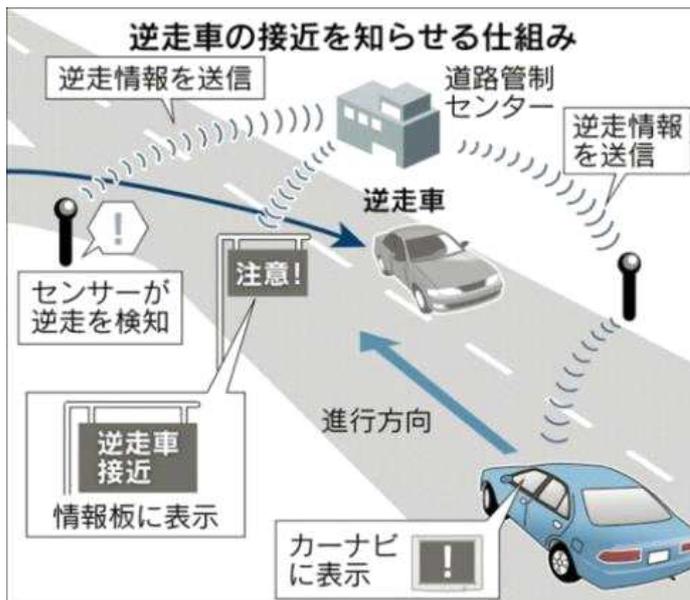


그림 6.7. 잘못된 방향 주행 경고의 예 (출처: 요미우리신문)

6.4.2.3. 추월 금지 경고



영국 왕립재해예방협회(RoSPA; Royal Society for the Prevention of Accidents)가 발간한 팩트시트(factsheet)에 따르면, 추월은 가장 위험한 운전 상황 중 하나로 간주되며, 도로 중앙 쪽에서 움직이는 차량별 추월하는 것이 2015년 전체 추월 관련 사고의 절반 이상을 차지했다[237]. 추월은 인간의 두뇌에서 매우 복잡한 추정과 의사 결정 과정뿐만 아니라 동작을 준비, 실행, 종료하는 동안 적절히 조정된 행동을 수반한다. 운전자가 추월 차선에서 사각지대나 느린 차량과 같은 안전하지 않은 상태를 인지하지 못한 경우, 추월 금지 경고(그림 6.8)는 생명을 구하는 정보가 될 수 있다. 자율 주행 자동차의 경우 이러한 경고는 차량 자체 센서의 범위를 효과적으로 확장하기 때문에 매우 유용하다.

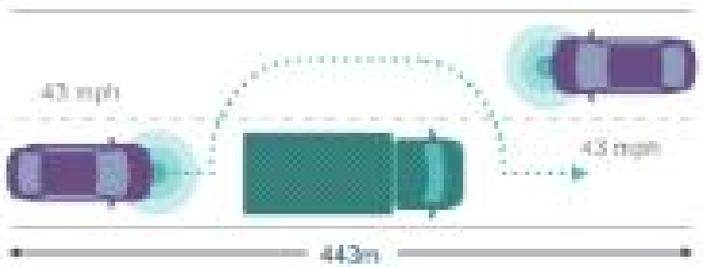


그림 6.8. 추월 금지 경고의 예(출처: 쉘컴)

6.4.3. V2P 통신을 이용한 자율주행 기술

6.4.3.1. VRU 경고

일반적으로, 운전자들은 보행자들이 차량을 보고 그 소리를 들을 수 있다고 가정해 차량의 존재를 자각할 수 있다고 가정하는 경향이 있다. 그러나 일부 보행자는 그러한 감각이 떨어지거나 시적으로 주의가 산만해질 수 있는데, 이는 보행자들이 도로를 건널 때 차량을 인식하는 것을 더 어렵게 만들 수 있다. 교차로나 횡단보도에서는 특히 보행자가 특별한 지각없이 건너도 안전하다고 생각할 수 있다. VRU(Vulnerable Road User) 경고는 이동성이 저하되거나 감각이 떨어지는 보행자가 도로를 건너려고 한다는 것을 차량에 알리는 데 사용할 수 있다. 자율 주행 자동차의 경우, 보행자를 감지하려고 라이더나 카메라와 같은 것에만 의존하는 것이 아니라 이러한 경고를 활용할 수 있다.

마찬가지로 VRU는 자전거 이용자의 존재를 차량에 알리는 데 사용될 수 있다. 자전거를 타는 사람들은 물리적 보호가 거의 없기 때문에 차 안에 있는 사람들보다 안전에 훨씬 더 취약하다. 또한 조명이 있음에도 불구하고 자전거 타는 사람들은 밤에 발견하기가 매우 어렵다. 왜냐하면, 그들은 연석에 가까이 있는 경향이 있고 배경과 겹쳐서 구분하기가 어렵기 때문이다. VRU를 사용해 자전거 이용자의 존재에 대해 경고하는 것은 자율 주행 자동차에게 유용할 것이다.

6.4.4. 협동 자율 운전

상호협력 자율주행은 협동감지와 상호협력 결정의 두 가지 범주로 나눌 수 있다[238]. 상호협력 센싱은 V2V와 V2I간의 감지 정보공유에 중점을 둔다. 이러한 데이터 공유는 자율 주행 차량의 감지 범위를 늘려 시스템을 더욱 견고하게 만든다. 협력적 결정을 통해 자율 주행 차량 그룹이 협력하고 결정을 내릴 수 있다.

일부 연구는 협동 자율주행을 위한 애플리케이션 탐색에 초점을 맞추고 있다. 문헌 [238에서는 호송운전, 협력 차선변경, 협력 교차로 관리, 협력감지 등 네 가지 사용 사례가 시연되었다. AutoNet2030[239]의 설계에 따르면, 호송은 다중 차선의 차량으로 그룹으로 구성되고 전체 그룹의 제어가 분산된다. 호송의 안전과 효율적인 제어를 위해서는 각 차량의 동적 데이터에 대한 고주파 교환이 필요하다. 그림 6.9에서 볼 수 있듯이 도로변 인프라는 교차로를 안전하게 통과하기 위해 차량을 조정하고 관리하는 데 사용된다. 문헌 [240]의 하나의 호송 제어 알고리즘은 호송의 모든 차량이 아닌 주변 차량의 동적 정보만 교환한다. 이 디자인은 알고리즘을 수렴하기 쉽게 만든다.



그림 6.9. 교차로에서의 V2X 통신(출처: 서울시 교통정보과)

협력 차선변경은 차선을 변경할 때 차량이 협력하도록 설계되었다. 적절한 협력 차선변경은 교통사고를 피할 수 있을 뿐만 아니라 교통 혼잡을 감소시킨다[241]. MOBIL은 차선 변경으로 인한 전반적인 제동을 최소화하는 것을 목적으로 하는 일반적인 모델이다[242]. 협동 교차로 관리는 안전한 운전과 교통 통제에도 도움이 된다. 효과적인 상호협력 교차점 관리는 차량 대 차량 및 차량 대 인프라 간의 조정 메커니즘을 기반으로 한다.



협력감지는 V2X 통신을 통해 자율주행 차량감지 범위를 늘린다. 한편, 협동감지는 자율주행 구축비용을 줄이는 데도 도움이 된다. 차량이 도로변 인프라에 배치된 센서에 더 많이 의존할 수 있으므로 차량용 센서 비용을 줄일 수 있다. 미래에는 센서 정보가 도로변 인프라에서 제공하는 차량에 대한 서비스가 될 수 있다.

V2X 네트워킹 인프라는 또한 협동 자율주행에 있어 매우 중요한 측면이다. 이기종 차량 네트워크(Heterogeneous Vehicular Network; HetVNET)[243]는 ITS의 통신 요구 사항을 충족하기 위한 네트워킹 인프라에 대한 초기 작업이다. HetVNET은 단일 무선 액세스 네트워크에 의존하는 이 동적 환경에서 만족스러운 서비스를 제공할 수 없기 때문에 LTE DSRC를 통합한다[244]. 문헌 [245]에서는 HetVNET에서 자율주행의 여러 응용 시나리오를 지원하기 위해 개선된 프로토콜 스택이 제안되었다. 프로토콜에서 저자들은 자율주행을 지원하기 위해 HetVNET의 제어 메시지를 재정의했다.

마찬가지로 차량용 지연허용 네트워크 VDTN(Vehicular Delay-Tolerant Network)[246]은 긴 지연과 산발적인 연결이 있는 시나리오를 위해 설계된 혁신적인 통신 아키텍처다. 아이디어는 메시지가 단거리 WiFi 연결로 전달되고 비동기적으로 대상에 도달할 수 있도록 하는 것이다. 이 속성을 통해 VDTN은 현재 VANET에 중단 간 경로가 없는 경우에도 서비스 및 응용 프로그램을 지원할 수 있다. 문헌 [247]에서 저자들은 VDTN에 대한 몇 가지 협력 전략을 논의했다. VDTN에서 협력의 과제는 제한된 대역폭, 에너지 자원 및 스토리지를 서로 공유하도록 차량 노드를 조정하는 방법이다. 또한, 협동 행동에 대해 차량에 보상하거나 처벌하는 인센티브 메커니즘이 제안된다.

원활한 V2X 통신을 지원하기 위해 핸드오버는 V2X 네트워킹 인프라에서도 매우 중요한 주제다. 네트워킹 토폴로지의 동적 변경과 상대적으로 통신 범위가 작으므로 셀룰러 네트워크의 핸드오버 메커니즘은 더는 VANET에 적합하지 않다. 선제적 자원 할당 기술을 기반으로 문헌 [248]에서 저자들은 VANET을 위한 새로운 핸드오버 모델을 제안한다. 사전 핸드오버의 도움으로 차량 이동과 함께 노변 기지국 RSU(Roadside Unit)를 통해 협력 서비스를 마이그레이션 할 수 있다. 따라서 안정적이고 효율적인 협력 시스템을 개발하려면 사전 핸드오버 및 자원 할당의 적절한 설계가 필수적이다.

자동차 산업에서 첨단 컴퓨팅의 발전도 매우 고무적이다. AECC (Automotive Edge Computing Consortium)는 미래 자동차에서 에지 컴퓨팅 기술을 홍보하기 위해 자동차 회사가 구성한 그룹이다(<https://aecc.org>), AECC에 따르면 서비스 시나리오에는 지능형 운전, 고화질 지도, V2Cloud 크루즈 지원, 금융 및 보험과 같은 일부 확장 서비스가 포함된다. 또한, 논문에서는 데이터 소스, 차량에서 생성되는 데이터 양, 목표 데이터 트래픽 속도, 응답 시간 및 필요한 가용성 측면에서 서비



스 요구사항에 대해 설명한다.

6.4.5. 남아있는 문제

해결해야 할 V2X의 주요 문제 중 하나는 네트워크 커버리지다. 일반적으로 V2X 네트워크는 국가나 지방자치 단체의 과감한 투자를 통해 기존 도로에서 RSU, 지능형 센서, 통신 백본 등을 갖춘 새로운 인프라로 업그레이드하기 때문에 해당하는 특정 지역에서만 사용할 수 있다. 또한 현재 시점에서 전 세계 대부분의 차량은 V2X를 지원하지 않는다. 이는 필수 사항이 아니기 때문이다. 2016년 말, 미국 NHTSA는 모든 신형 장치가 DSRC 기반 V2V 통신을 지원하도록 요구하는 명령을 제안했다[249]. 그러나 현재로서는 해당 제안이 불확실해 보인다[250].

해결해야 할 또 다른 큰 과제는 상호 운용성 문제다. 역사적으로 V2X 통신 표준은 전 세계의 다른 표준화 기구, 특히 미국과 유럽에서 동시에 개발됐다. 이러한 병렬적인 표준 개발은 서로 완전히 호환되지 않는 일련의 다른 표준으로 이어졌다(지금도 미국은 지역별로 다른 셀룰러 주파수를 사용한다). 이는 디지털 휴대폰의 초기 시대에 경험했던 상황과 비슷하다고 할 수 있다. 2009년 11월, 유럽 위원회와 USDOT(U.S. Department of Transportation)의 대표들은 미국과 유럽 ITS 표준 간의 협력 과정이 시작됐음을 알리는 ‘협동적 시스템에서의 연구 협력에 대한 공동 의향 선언’에 서명했다[251]. 그러나 현재 V2X 조각화의 주된 이유인 C-V2X 표준의 존재로 인해 수립된 표준이 전체 V2X 생태계를 수립하기에 충분하지 않을 수 있다.

6.5. 백엔드 시스템

아무리 정교한 하드웨어와 지능형 소프트웨어가 설치되더라도 자율 주행차가 외부에서 인식하는 것에는 여전히 한계가 있다. 이 제한은 센서의 최대 작동 범위, 센서 폐쇄, 악천후 등으로 인해 발생할 수 있다. 따라서 자율 주행차가 스스로 인식할 수 있는 것 이상의 외부 정보를 얻는 것이 때로는 필수적일 수 있으므로 미리 계획하고 좀 더 시기 적절하게 더 나은 결정을 내려야 한다. 차량 대 차량(V2V) 통신 또는 차량 대 인프라(V2I) 통신도 이 문제를 해결할 수 있지만, 자율 주행차가 다른 V2V 지원 차량이나 V2I 지원 인프라에 근접한 경우에만 이러한 통신을 사용할 수 있는 한계가 있다.

또 다른 중요한 이유는 실시간 교통 정보나 실시간 지도 업데이트가 필요하다는 점이다. 차량에 저장된 일부 내부 정보가 더 이상 정확하지 않아서 실시간으로 업데이트해야 할 수 있다. 차선 임시 폐쇄나 사고 또는 공사로 인한 역방향 통행 등이 그 예다. 백엔드 서버는 다양한 소스에서 정보를 수집해 각 SDV에 이러한



관련 정보를 전달할 수 있다.

사용 가능한 컴퓨팅 리소스에 따라 일부 사용량이 많거나 리소스가 많이 사용되는 작업을 백 엔드 시스템으로 오프로드해야 하고 계산 결과를 자율 주행차로 다시 전송해야 하는 경우가 발생할 수 있다. 이러한 접근 방식은 스마트폰에서 흔히 볼 수 있다. 일반적으로 이러한 작업에는 많은 데이터를 처리해야 하는 비안전 계산 작업, 자율 주행차에서 로컬로 사용할 수 없는 데이터를 사용하는 작업, 백 엔드 서버에서 좀 더 효율적으로 수행할 수 있는 작업 등이 있다.

백엔드 시스템은 차량 관리 작업에도 필수적이다. 자율주행 공공 셔틀을 예로 들어보면, 백엔드 시스템은 자동으로 노선의 일정을 변경하거나 실시간 상황에 따라 특정 셔틀을 충전소로 보낼 수 있다. 또한 예비 셔틀은 현재 대기 인원이 많을 때 자동으로 투입할 수 있다. 또한 자율 주행차 운영자나 제조업체는 원격 진단 수행, 차량 상태 실시간 모니터링, 백엔드 시스템을 통한 OTA(Over-The-Air) 소프트웨어 업데이트 등을 실행할 수 있다.

6.5.1. 백엔드 시스템 기능

6.5.1.1. SOTA 업데이트

차량 소프트웨어를 업데이트하는 전통적이고 일반적인 방법은 수리점을 방문하는 동안 차량에 부착된 특수한 도구를 통해 이뤄진다. SOTA(Software Over The Air)는 공용 셀룰러 네트워크 또는 개인/공용 Wi-Fi 핫스팟과 같은 일반적인 모바일 네트워크를 사용해 원격으로 차량 소프트웨어 업데이트를 수행할 수 있는 방법이다. SOTA는 소비자 가전 소프트웨어 업데이트를 수행하기 위한 확립된 방식이며, 자동차 제조업체는 SOTA 아이디어를 수용하고 SOTA를 대체 소프트웨어 업데이트 메커니즘으로 지원하고자 적극적으로 노력하고 있다.

6.5.1.2. HD 지도

SDV는 지도를 사용해 위치를 파악한다. 정확한 위치 파악을 위해 SDV는 HD 지도와 같은 매우 높은 정밀도(종종 센티미터 수준의 정확도) 수준의 정확도의 운영 환경 지도가 필요하다. 그러나 그림 6.10에 나온 것처럼 HD 지도는 엄청난 양의 데이터를 저장해야 한다. 자율 주행차 컴퓨팅 플랫폼의 사용 가능한 하드웨어 리소스와 자율 주행차 운영 영역의 크기에 따라 모든 지도 데이터를 차량에 저장하지 못할 수 있다. 따라서 자율 주행차는 필요에 따라 백엔드 서버에서 새로운 지도 데이터나 누락된 지도 데이터를 요청할 수 있다. 백엔드 서버는 또한 내부 지도를 업데이트하거나 도로 공사로 인한 차선 폐쇄, 교통사고 등 경로 관련 이벤트에 대한 알림을 보낼 수 있다.

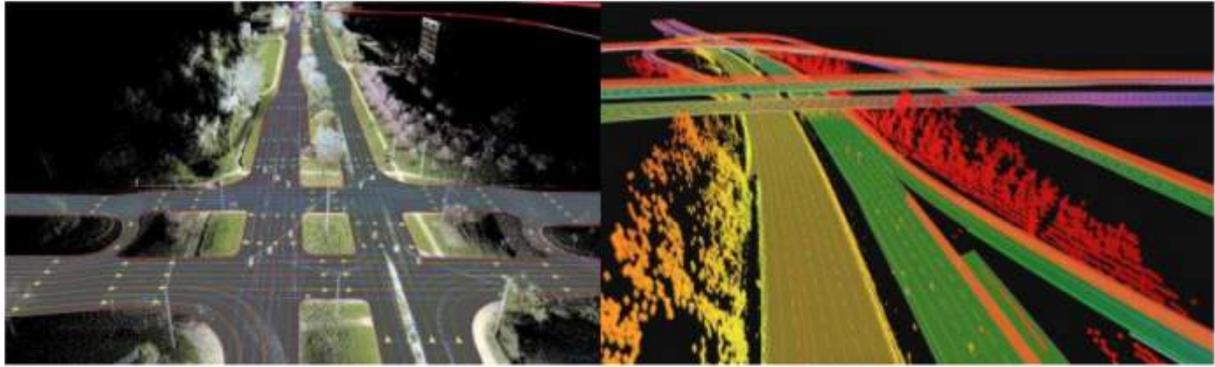


그림 6.10. HD 지도의 예 (출처: [252])

6.5.1.3. 차량 관리

자율주행 공공 셔틀이나 라스트 마일 배송 시스템(last mile delivery system)과 같은 일부 자율 주행 적용사례에서는 여러 자율 주행차가 함께 작동해 서비스를 제공한다. 차량 관리(fleet management) 서비스는 모든 자율 주행차의 원활하고 안전하며 효율적인 작동을 보장한다. 일반적인 차량 관리 서비스에는 그림 6.11과 같이 각 자율 주행차의 위치 추적, 서비스 전송, 동적 경로 계산, 시스템 상태 모니터링, 원격 진단이 포함된다. 차량관리 서비스는 관제실에서 작업자가 수동으로 수행하거나 백엔드 서버에서 차량 관리 소프트웨어를 사용해 자동화할 수 있으며, 두 가지 모두를 조합해 수행할 수도 있다.

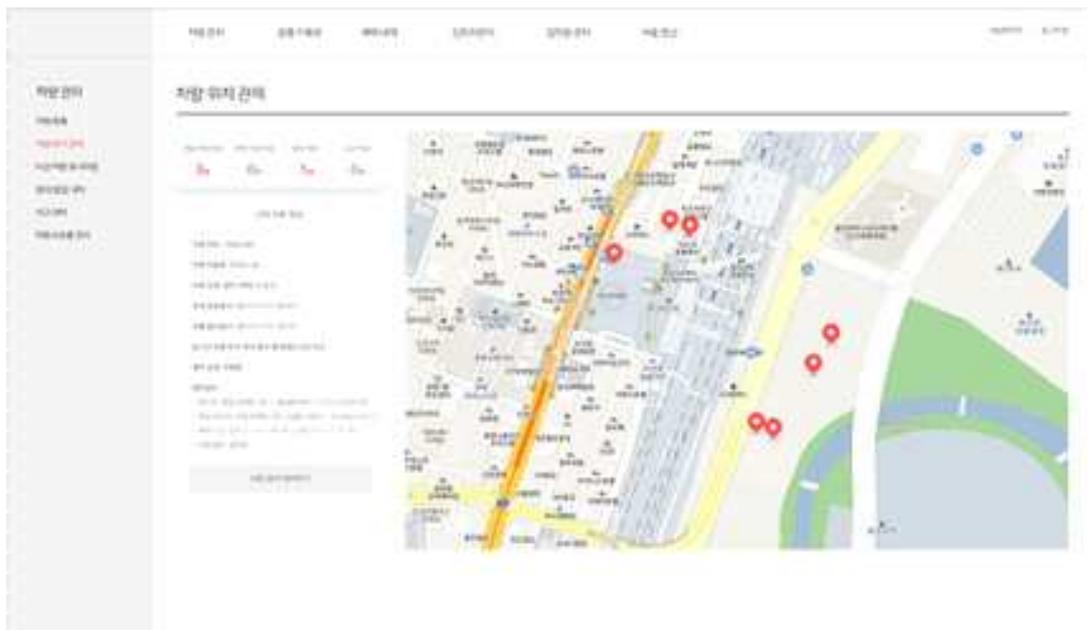


그림 6.11. 차량 관리 소프트웨어의 예(출처: 휴비넷)

6.5.1.4. 남아있는 문제



백엔드 시스템의 목적은 자율 주행차가 원활하고 안전하면서 안정적으로 구동 되도록 보장하는 것이지만, 전체 자율 주행 시스템에 백엔드 시스템을 추가하기 전에 신중하게 해결해야 하는 몇 가지 문제가 있다. 첫 번째 문제는 외부 공격으로부터 시스템을 보호하고 데이터 프라이버시를 보장하는 사이버 보안이다. 전체 시스템 보안은 제조업체 또는 운영자가 외부 통신을 얼마나 잘 보호하느냐에 따라 효과적으로 결정된다. 또 다른 문제는 사용되는 통신 네트워크의 QoS(서비스 품질)이다(예: 지연 시간, 처리량, 손실). 자율 주행차와 백엔드 서버 간에 교환하는 데이터의 보안이 얼마나 중요한지에 따라 시설 전용 무선 네트워크를 사용하는 것이 공용 무선 네트워크를 사용하는 것보다 더 나을 수 있지만 이는 설치 및 유지 관리 비용이 상당히 높아진다.

자율 주행차와 백엔드 시스템 간 통신 프로토콜의 독점적 특성은 제조업체 (또는 OEM)가 교환할 수 있는 데이터의 종류와 빈도를 완벽하게 제어할 수 있도록 한다. 개인 정보 보호 관점에서 유리할 수 있지만(데이터가 다른 당사자와 공유되지 않기 때문에), 다른 도로 참여자들의 활용 가능한 잠재적 정보에 대한 접근을 제한한다는 한계점이 있다. 또 다른 문제는 개별 백엔드 서비스의 적용 범위다. 엄청난 양의 데이터로 인해 HD 지도는 특정 지역에서만 사용할 수 있다. 그렇기 때문에 자율 주행차는 해당 지역에 대한 HD 지도가 오래됐거나 아직 사용할 수 없는 상태일 경우 새로운 지역에서 위치를 파악하는데 어려움을 겪을 수 있다.



7. 결론

7.1. 사회적 전망

자율 주행 기술의 광범위한 사용은 임박하지 않았다. 그러나 어느 정도 잠재적인 영향과 이점을 예측하는 것은 여전히 가능하다.

- 1) 해결할 수 있는 문제: 교통사고 예방, 교통 혼잡 완화, 배출 감소
- 2) 기회 증가: 운전 시간 재할당, 이동 장애인 이동
- 3) 새로운 트렌드: MaaS(Mobility as a Service) 소비, 물류 혁명

자율 주행 기술의 광범위한 배포는 주의 산만, 음주 운전 및 과속과 같은 잘못된 인간 행위로 인한 사회적 손실을 줄일 수 있다[3]. 세계적으로, 노년층(60세 이상)이 젊은 그룹보다 빠르게 성장하고 있다[253]. 노인들의 이동성을 증가시키는 것은 인구의 많은 부분의 삶의 질과 생산성에 큰 영향을 미칠 수 있으며, 이를 가능하게 하는 것이 자율 주행 시스템이다.

개인 차량 소유에서 MaaS(Mobility as a Service)를 소비하는 방향으로의 전환은 새로운 추세이다. 현재 승차 공유는 연간 주행 거리가 1000km 미만인 차량 소유에 비해 비용이 저렴하다[254]. 공유 차량 대비 소유 비율은 2030년까지 50:50이 될 것으로 예상되며[255], 자율 주행차의 대중화는 이러한 추세를 가속화할 수 있다.

7.2. 과제

자율 주행차는 불확실한 환경에서 작동하는 복잡한 시스템이다. 따라서, 미해결 문제가 있는 시나리오는 무수히 많다.

SAE(Society of Automotive Engineers)는 5가지 수준의 운전 자동화를 정의했다 [19]. 이 분류법에서 레벨 0은 자동화를 전혀 의미하지 않는다. 어댑티브 크루즈 컨트롤, ABS(안티 브레이크 시스템) 및 스테빌리티 컨트롤과 같은 기본 운전자 보조 시스템은 레벨 1부터 시작한다[256]. 레벨 2는 비상 제동 또는 충돌 회피와 같은 첨단 보조 시스템[257, 258]이 통합된 부분 자동화이다. 차량 제어 분야의 축적된 지식과 업계의 경험으로 레벨 2 자동화는 실현 가능한 기술이 되었다. 진정한 도전은 이 수준 이상에서 시작된다.

레벨 3은 조건부 자동화로, 운전자는 정상 작동 중 주행 이외의 작업에 집중할 수 있지만, 차량의 비상 경보에 신속하게 대응하고 인계받을 준비를 해야 한다. 또한 레벨 3 자율 주행차는 고속도로와 같은 제한된 ODD(Operational Design



Domain)에서만 작동한다. 아우디는 제한된 고속도로 조건에서 레벨 3 자동화를 달성한 최초의 생산 자동차라고 주장한다[259]. 그러나 운전자가 자동 모드에서 수동으로 제어권을 넘겨받는 것은 또 다른 문제를 제기한다. 최근의 연구 [260, 261]는 이 문제를 조사하였고 인수 상황이 주변 차량과의 충돌 위험을 증가시킨다는 것을 발견했다. 인수 과정에서 사고가 날 가능성이 높아진 것은 아직 해결되지 않은 문제다.

레벨 4와 레벨 5에서는 인간의 관심이 전혀 필요하지 않다. 그러나 레벨 4는 특수 인프라 또는 상세 지도가 존재하는 제한된 ODD에서만 작동할 수 있다. 이러한 구역에서 출발하는 경우, 차량은 자동으로 주차하여 이동을 중지해야 한다. 완전 자동화된 시스템인 레벨 5는 어떠한 도로망과 날씨 조건에서도 작동할 수 있다. 아직 레벨 4 또는 레벨 5 주행 자동화가 가능한 생산 차량은 없다. 게다가, 도요타 연구소는 업계에서 레벨 5 자동화에 근접한 사람은 아무도 없다고 밝혔다 [262].

도시 도로 네트워크에서 4단계 이상의 자동화는 개방적이고 도전적인 문제이다. 기상 조건에서부터 주변 인간의 행동에 이르기까지 환경 변수는 매우 불확실하고 예측하기 어렵다. 또한 시스템 고장으로 인해 사고가 발생하는데, 현대 대회에서는 자율 주행차 중 한 대가 비로 인해 추락했고[263], 구글의 자율 주행차는 버스의 속도를 추정하지 못해 차선을 매달던 버스를 들이받았으며[264], 테슬라의 오토파일럿은 흰색 트럭을 인식하지 못해 충돌하여 운전자가 사망했다[265].

미성숙한 기술로 인한 사망[265, 266]은 자율 주행차에 대한 대중의 수용을 저해한다. 최근 조사[254]에 따르면, 대다수의 소비자들은 기술의 안전성에 의문을 제기하며, 자율 주행차의 개발 및 사용에 대해 정부가 통제를 하기 원한다. 반면에, 자율 주행차에 대한 극도로 신중한 접근도 부정적인 인상을 주고 있다[267].

윤리적 딜레마는 또 다른 도전을 제기한다. 불가피한 사고 상황에서, 시스템은 어떻게 동작해야 하는가[268]? 이 문제와 관련하여 실험윤리가 제안되었다[269].

리스크 및 신뢰성 인 증은 아직 해결해야 할 또 다른 과제다. 항공기와 마찬가지로 ADS는 치명적인 고장의 가능성을 최소화할 수 있는 높은 중복성으로 설계되어야 한다. 이와 관련하여 DeepTest[270]와 같은 유망한 프로젝트가 있음에도 불구하고, 설계-시뮬레이션-시험-설계 재인증 절차는 여전히 업계나 규칙 제정자에 의해 확립되지 않고 있다.

마지막으로, 목적지까지 도달하는 시간, 연비, 편안함, 승차 공유 최적화와 같은 다양한 최적화 목표는 이미 해결하기 어려운 문제의 복잡성을 증가시킨다. 따라서 잘 정의된 지오펜스(Geo-fence) 영역 밖의 엄격한 조건하에서 모든 동적 주행 작업을 안전하게 수행하는 것은 여전히 미해결 문제로 남아 있다.

참고 문헌

- [1] Singh, S. Critical reasons for crashes investigated in the national motor vehicle crash causation survey. National Center for Statistics and Analysis, 2018.
- [2] Crayton, T. J. and Meier, B. M. Autonomous vehicles: Developing a public health research agenda to frame the future of transportation policy. *Journal of Transport & Health*, 2017, 6, pp. 245-252.
- [3] Montgomery, W. D., Mudge, R., Groshen, E. L., Helper, S., MacDuffie, J. P., and Carson, C. America's workforce and the self-driving future: Realizing productivity gains and spurring economic growth. 2018.
- [4] Lavrinc, D. This is how bad self-driving cars suck in rain. 2014. <https://jalopnik.com/this-is-how-bad-self-driving-cars-suck-in-the-rain-1666268433>.
- [5] Davies, A. Google's self-driving car caused its first crash. 2016. <https://www.wired.com/2016/02/googles-self-driving-car-may-caused-first-crash/>.
- [6] Lee, T. B. Autopilot was active when a tesla crashed into a truck, killing driver. <https://arstechnica.com/cars/2019/05/feds-autopilot-was-active-during-deadly-march-tesla-crash/>.
- [7] Thorpe, C., Herbert, M., Kanade, T., and Shafter, S. Toward autonomous driving: the cmu navlab. ii. architecture and systems. *IEEE Expert*, 1991, 6(4), 44-52.
- [8] Broggi, A., Bertozzi, M., & Fascioli, A. Argo and the millemiglia in automatico tour. *IEEE Intelligent Systems and their Applications*, 1999, 14(1), 55-64.
- [9] Gregor, R., Lutzeler, M., Pellkofer, M., Siedersberger, K.-H., and Dickmanns, E. D. Ems-vision: A perceptual system for autonomous vehicles. *IEEE Transactions on Intelligent Transportation Systems*, 2002, 3(1), 48-59.
- [10] Stuart, J. and Sanders, D. European Eureka project: Programme for a european traffic system with highest efficiency and unprecedented safety. <https://researchportal.port.ac.uk/en/projects/european-eureka-project-programme-for-a-european-traffic-of-highe>.
- [11] Ulmer, B. Vita II-active collision avoidance in real traffic. In *Proceedings of the Intelligent Vehicles '94 Symposium*, 1994, pp. 1-6.
- [12] Buehler, M., Iagnemma, K., and Singh, S. The 2005 DARPA grand



challenge: the great robot race. Springer, 2007.

[13] Buehler, M., Iagnemma, K., and Singh, S. The DARPA urban challenge: autonomous vehicles in city traffic. Springer, 2009.

[14] Broggi, A., Cerri, P., Felisa, M., Laghi, M. C., Mazzei, L., and Porta, P. P. The vislab intercontinental autonomous challenge: an extensive test for a platoon of intelligent vehicles. *International Journal of Vehicle Autonomous Systems*, 2012, vol. 10, no. 3, pp. 147-164.

[15] Broggi, A., Cerri, P., Debattisti, S., Laghi, M. C., Medici, P., Molinari, D., Panciroli, M., and Prioletti, A. Proud-public road urban driverless-car test. *IEEE Transactions on Intelligent Transportation Systems*, 2015, 16(6), pp. 3508-3519,

[16] Cerri, P., Soprani, G., Zani, P., Choi, J., Lee, J., Kim, D., Yi, K., and Broggi, A. Computer vision at the hyundai autonomous challenge. In *14th International Conference on Intelligent Transportation Systems (ITSC)*. IEEE, 2011, pp. 777-783.

[17] Englund, C., Chen, L., Ploeg, J., Semsar-Kazerooni, E., Voronov, A., Bengtsson, H. H., and Didoff, J. The grand cooperative driving challenge 2016: boosting the introduction of cooperative automated vehicles. *IEEE Wireless Communications*, 2016, 23(4), pp. 146-152.

[18] SAE (2018). Taxonomy and definitions for terms related to driving automation systems for on-road motor vehicles: Technical report, Warrendale, PA, USA: SAE International.

[19] Liu, S., Tang, J., Zhang, Z., and Gaudiot, J.-L. Computer architectures for autonomous driving. *Computer*, 2017, 50(8), pp. 18-25.

[20] Misra, P. and Enge, P. 2006. *Global Positioning System: Signals, Measurements and Performance*, 2nd ed. MA: Ganga-Jamuna Press. 15

[21] Leick, A., Rapoport, L., and Tatarnikov, D. *GPS Satellite Surveying*. John Wiley & Sons, 2015.

[22] Zumberge, J. F., Hefiin, M.B., Jefferson, D.C., Watkins, M.M., and Webb, F.H. Precise point positioning for the efficient and robust analysis of GPS data from large networks. *Journal of Geophysical Research: Solid Earth*, 1997, 102(B3), pp. 5005- 5017.

[23] Gao, y. and Chen, K. Performance analysis of precise point positioning using real-time orbit and clock products. *Journal of Global Positioning Systems*, 2004, 3(1- 2), pp. 95-100.

[24] Collins, J.p. and Langley, R.B. A Tropospheric Delay Model for the User of



the Wide Area Augmentation System. Department of Geodesy and Geomatics Engineering, University of New Brunswick, 1997.

[25] Wubben, G., Schmitz, M., and Bagge, A. September. PPP-RTK: precise point positioning using state-space representation in RTK networks. In Proceedings of IONGNSS, 2005, 5, pp. 13-16.

[26] Geng, J., Teferle, F.N., Meng, X., and Dodson, A.H. Towards PPP-RTK: Ambiguity resolution in real-time precise point positioning. Advances in Space Research, 2011, 47(10), pp. 1664-1673.

[27] Laurichesse, D. September. The CNES Real-time PPP with undifferenced integer ambiguity resolution demonstrator. In Proceedings of the ION GNSS, 2011, pp. 654-662.

[28] Grinter, T. and Roberts, C. Real time precise point positioning : Are we there yet? IGSS Symposium, 2013.

[29] Caron, F., Dufios, E., Pomorski, D., and Vanheeghe, P. GPS/IMU data fusion using multisensor Kalman filtering: introduction of contextual aspects. Information Fusion, 2006, 7(2), pp. 221-230.

[30] Schwarz, B. LIDAR: Mapping the world in 3D. Nature Photonics, 2010, 4(7), p.429.

[31] Muhammad, N. and Lacroix, S. October Calibration of a rotating multibeam lidar. In 2010 IEEE/RSJ International Conference on Intelligent Robots and Systems(IROS), 2010, pp. 5648-5653.

[32] Montemerlo, M. and Thrun, S. Large-scale robotic 3-d mapping of urban structures. Experimental Robotics IX, 2006, pp. 141-150.

[33] Levinson, J., Montemerlo, M., and Thrun, S. June. Map-based precision vehicle localization in urban environments. In Robotics: Science and Systems, 2008, pp 121-128.

[34] Konolige, K. Large-scale map-making. In AAAI, 2004, pp. 457-463.

[35] Liu, S., Li, L., Tang, J. et al. Creating autonomous vehicle systems. Synthesis Lectures on Computer Science, 2017, 6 (1), pp. 1-186.

[36] Liu, S., Tang, J., Wang, C. et al. A unified cloud platform for autonomous driving. Computer, 2017, 50(12), pp. 42-49.

[37] Thrun, S. and Montemerlo, M. The graph SLAM algorithm with applications to large-scale mapping of urban structures. The International Journal of Robotics Research, 2006, 25(5-6), pp. 403-429.

[38] Yu, Y., Li, J., Guan, H. et al. Semiautomated extraction of street light



poles from mobile LiDAR point-clouds. *IEEE Transactions on Geoscience and Remote Sensing*, 2015, 53(3), pp.1374-1386.

[39] Zheng, H., Wang, R., and Xu, S. Recognizing street lighting poles from mobile LiDAR data. *IEEE Transactions on Geoscience and Remote Sensing*, 2017, 55(1), pp. 407-420.

[40] Ordonez, C., Cabo, C., and Sanz-Ablanedo, E. Automatic detection and classification of pole-like objects for urban cartography using mobile laser scanning data. *Sensors*, 2017, 17(7), pp. 1-10.

[41] Fukano, K. and Masuda, H. Detection and classification of pole-like objects from mobile mapping data. In *Proceedings of ISPRS Annals of Photogrammetry, Remote Sensing & Spatial Information Sciences*, 2015, pp. 57-64.

[42] Qi, C. R., Su, H., Mo, K., and Guibas, L.J. PointNet: deep learning on point sets for 3D classification and segmentation. In *Proceedings of 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 77-85.

[43] Qi, C. R., Yi, L., Su, H., and Guibas, L.J.. PointNet++: Deep Hierarchical Feature Learning on Point Sets in a Metric Space. In *Proceedings of Advances in Neural Information Processing Systems 30 (NIPS 2017)*, pp. 1-14.

[44] Zhou, Y. and Tuzel, O. VoxelNet: End-to-End Learning for Point Cloud Based 3D Object Detection. In *Proceedings of in 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018, pp. 4490-4499.

[45] Levinson, J. and Thrun, S. Robust vehicle localization in urban environments using probabilistic maps. In *Proceedings of IEEE/RSJ International Conference on Robotics and Automation(ICRA 2010)*, 2010, pp. 4372-4378.

[46] Baldwin, I. and Newman, P. Road vehicle localization with 2d pushbroom lidar and 3d priors. In *Proceedings of IEEE International Conference on Robotics and Automation (ICRA 2012)*, 2012, pp. 2611-2617.

[47] Chong, Z.J., Qin, B., Bandyopadhyay, T., Ang, M.H., Frazzoli, E., and Rus, D. Synthetic 2d lidar for precise vehicle localization in 3d urban environment. In *Proceedings of IEEE International Conference on Robotics and Automation (ICRA 2013)*, 2013, pp. 1554-1559.

[48] Wolcott, R.W. and Eustice, R.M. Fast LIDAR localization using multiresolution Gaussian mixture maps. In *Proceedings of IEEE International Conference on Robotics and Automation(ICRA 2015)*, 2015, pp. 2814-2821.

[49] Chen, Z. Bayesian filtering: From Kalman filters to particle filters, and



beyond. *Statistics*, 2003, 182(1), pp. 1-69.

[50] 전슬기, 성창기, 명현. 3D LiDAR를 이용한 차선 검출 및 정밀지도 매칭을 통한 자율주행 자동차의 위치 인식 기법. *한국통신학회 학술대회논문집*, 2020, pp. 347-348.

[51] Scaramuzza, D. and Friedrich, F. Visual odometry [tutorial]. *Robotics & Automation Magazine*, IEEE, 2011, 18(4), pp. 80-92.

[52] Nistér, D., Naroditsky, O., and Bergen, J. Visual odometry for ground vehicle applications. *Journal of Field Robotics*, 2006, 23(1), pp. 3-20.

[53] Forster, C., Pizzoli, M., and Scaramuzza, D. SVO: Fast semi-direct monocular visual odometry. In *Proceedings of IEEE International Conference on Robotics and Automation(ICRA 2014)*, 2014, pp. 15-22.

[54] Nister, D. An efficient solution to the five-point relative pose problem.' *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2004, 26(6), pp. 756-770.

[55] Fischler, M. A. and Firschein, O. *Readings in Computer Vision: Issues, Problems, Principles, and Paradigms*. Morgan Kaufmann, 1987.

[56] Ruppelt, J. and Trommer, G. F. Stereo-camera visual odometry for outdoor areas and in dark indoor environments. *IEEE Aerospace and Electronic Systems Magazine*, 2016, 31(11), pp. 4-12.

[57] Howard, A. Real-time stereo visual odometry for autonomous ground vehicles. In *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems(IROS 2008)*, 2008, pp. 3946-3952.

[58] Li, M. and Mourikis, A. I. High-precision, consistent EKF-based visual-inertial odometry. *The International Journal of Robotics Research*, 2013, 32(6), pp. 690-711.

[59] Mourikis, A. I. and Roumeliotis, S. I. A multi-state constraint Kalman filter for vision-aided inertial navigation. In *Proceedings 2007 IEEE International Conference on Robotics and Automation*, 2007, pp. 3565-3572.

[60] Kitt, B. M., Rehder, J., Chambers, A. D., Schonbein, M., Lategahn, H., and Singh, S. Monocular visual odometry using a planar road model to solve scale ambiguity. In *Proceedings of 5th European Conference on Mobile Robots (ECMR '11)*, 2011, pp. 1-6.

[61] Yin, X., Wang, X., Du, X., and Chen, Q. Scale recovery for monocular visual odometry using depth estimated with deep convolutional neural fields. In *Proceedings of IEEE International Conference on Computer Vision (ICCV 2017)*,



2017, pp. 5871-5879.

[62] Leutenegger, S., Lynen, S., Bosse, M., Siegwart, R., and Furgale, P. Keyframe-based visual-inertial odometry using nonlinear optimization. *The International Journal of Robotics Research*, 2015, 34(3), pp.314-334.

[63] Urmson, C., Anhalt, J., Bagnell, D., Baker, C., Bittner, R., Clark, M.N., Dolan, J., Duggins, D., Galatali, T., Geyer, C., and Gittleman, M. Autonomous driving in urban environments: Boss and the urban challenge. *Journal of Field Robotics*, 2008, 25(8), pp. 425-466.

[64] Monemerlo, M., Becker, J. Bhat, S., Dahlkamp, H., Dolgov, D., Ettinger, S., Haehnel, D., Hilden, T., Hoffmann, G., Huhnke, B., and Johnston, D. Junior: The Stanford entry in the urban challenge. *Journal of Field Robotics*, 2008, 25(9), pp. 569-597.

[65] Zeigler, J., Bender, P., Schreiber, M., Lategahn, H. Strauss, T., Stiller, C., Dang, T., Franke, U., Appenrodt, N., Keller C.G., and Kaus, E. Making Bertha drive- An autonomous journey on a historic route *IEEE Intelligent Transportation Systems Magazine*, 2014, 6(2), pp. 8-20.

[66] Urmson, C., Baker, C., Dolan, J., Rybski, P., Salesky, B., Whittaker, W., Ferguson, D., and Darms, M. Autonomous Driving in Traffic: Boss and the Urban Challenge. *AI Magazine*, 2009, 30, pp. 17-28.

[67] Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., and Fei-Fei, L. Imagenet: A large-scale hierarchical image database. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition(CVPR 2009)*, 2009, pp. 248-255.

[68] Everingham, M., Van Gool, L., Williams, C. K., Winn, J., and Zisserman, A. The pascal visual object classes (voc) challenge. *International Journal of Computer Vision*, 2010, 88(2), pp. 303-338.

[69] Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollar, P., and Zitnick, C.L. September. Microsoft coco: Common objects in context. In *European Conference on Computer Vision*, 2014, pp. 740-755.

[70] Baker, S., Scharstein, D. , Lewis, J.p. , Roth, S., Black, M.J., and Szeliski, R. A database and evaluation methodology for optical flow. *International Journal of Computer Vision*, 2011, 92(1), pp. 1-31.

[71] Janai, J., Guney, F. , Wulff, J., Black, M., and Geiger, A. Slow flow: Exploiting highspeed cameras for accurate and diverse optical flow reference data. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition(CVPR 2017)*, 2017, pp. 1406-1416.



[72] Scharstein, D., Szeliski, R., and Zabih, R. A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. In Proceedings of IEEE Workshop on Stereo and Multi-Baseline Vision(SMBV 2001), 2001, pp. 131-140.

[73] Scharstein, D., Hirschmuller, H., Kitajima, y., Krathwohl, G., Nescic, N., Wang, x., and Westling, P. High-resolution stereo datasets with subpixel-accurate ground truth. In Proceedings of German Conference on Pattern Recognition, 2014, pp. 31-42.

[74] Leal-Taixe, L., Milan, A., Reid, I., Roth, S., and Schindler, K. 2015. MOTChallenge 2015: Towards a benchmark for multi-target tracking. arXiv preprint arXiv: 1504.01942. 52

[75] Milan, A., Leal-Taixé, L., Reid, I., Roth, S., and Schindler, K. 2016. Mot16: A

benchmark for multi-object tracking. arXiv preprint arXiv :1603.00831. 52

[76] Geiger, A., Lenz, P., and Urtasun, R. 2012. June. Are we ready for autonomous driving? the kitti vision benchmark suite. In 2012 IEEE Conference on Computer Vision and Pattern Recognition(CVPR), (pp. 3354-3361). IEEE. DOI: 10.1109/CVPR.2012.6248074. 52, 53

[77] Cordts, M., Omran, M., Ramos, S., Rehfeld, T., Enzweiler, M., Benenson, R., Franke, U., Roth, S., and Schiele, B. The cityscapes dataset for semantic urban scene understanding. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 3213-3223.

[78] Geiger, A., Lenz, P., Stiller, C., and Urtasun, R. Vision Meets Robotics : the KITTI Dataset. International Journal of Robotics Research, 2013, 32(11) pp. 1231-1237.

[79] Dalal, N. and Triggs, B. Histograms of oriented gradients for human detection. In Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition(CVPR 2005), 2005, 1, pp. 886-893.

[80] Felzenszwalb, P., McAllester, D., and Ramanan, D. A discriminatively trained, multiscale, deformable part model. In Proceedings of IEEE Conference on Computer Vision and Pattern Recognition(CVPR 2008), 2008, pp. 1-8.

[81] Li Fei - Fei Object recognition. https://vision.stanford.edu/documents/Fei-Fei_ICVSS07_ObjectRecognition_web.pdf.

[82] Lowe, D.G. Object recognition from local scale-invariant features. In Proceedings of the Seventh IEEE International Conference on Computer vision, 1999. 2, pp. 1150-1157.



- [83] Matas, J., Chum, O., Urban, M., & Pajdla, T. Robust Wide Baseline Stereo from Maximally Stable Extremal Regions. *Image and Vision Computing*, 2002, 22(10), pp. 761-767.
- [84] Salahat, E. and Qasaimeh, M. Recent advances in features extraction and description algorithms : A comprehensive survey. In *Proceedings of 2017 IEEE International Conference on Industrial Technology(ICIT2017)*, 2017, pp. 1059-1063.
- [85] Kim, J., Park, J., Chung, W. Self-Diagnosis of Localization Status for Autonomous Mobile Robots. *Sensors* 2018, 18, pp. 1-16.
- [86] Breiman, L. Random forests. *Machine Learning*, 2001, 45(1), pp. 5-32.
- [87] Louppe, G. Understanding random forests: From theory to practice. PhD dissertation, University of Liège, 2014.
- [88] 이형주, 정건희. 랜덤포레스트를 이용한 대설피해액에 대한 범주형 예측 및 개선방안 검토. *한국습지학회지*, 2019, 21(2), pp. 157-162.
- [89] Werbos, P. Beyond regression : new tools for prediction and analysis in the behavioral sciences Ph. D. dissertation, Harvard University, 1974.
- [90] He, X., Zemel, R.S., and Carreira-Perpinan, M.A. Multiscale conditional random fields for image labeling. In *Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition(CVPR 2004)*, 2004, 2, pp. 695-703.
- [91] He, X. , Zemel, R., and Ray, D. Learning and incorporating top-down cues in image segmentation. In *Proceedings of the 9th European conference on Computer Vision(ECCV 2006)*, 2006, pp. 338-351.
- [92] Krähenbühl, P. and Koltun, V. Efficient inference in fully connected crfs with gaussian edge potentials. In *Advances in Neural Information Processing Systems*, 2011, pp. 109-117.
- [93] Ladicky, L., Russell, C., Kohli, P., and Torr, P.H. Graph cut based inference with co-occurrence statistics. In *Proceedings of European Conference on Computer Vision*, 2010, pp. 239-253.
- [94] Bay, H., Tuytelaars, T., and Van Gool, L. Surf: Speeded up robust features. *Computer Vision(ECCV2006)*, 2006, pp. 404-417.
- [95] Hirschmuller, H. Stereo processing by semiglobal matching and mutual information. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2008, 30(2), pp. 328-341.
- [96] Drory, A., Haubold, C., Avidan, S., and Hamprecht, F.A. Semiglobal matching: a principled derivation in terms of message passing. In *Proceedings of*



German Conference on Pattern Recognition, 2014, pp. 43-53.

[97] Gehrig, S.K., Eberli, F., and Meyer, T. A real-time low-power stereo vision engine using semi-global matching. In Proceedings of International Conference on Computer Vision Systems, 2009, pp. 134-143.

[98] Horn, B.K. and Schunck, B.G. Determining optical flow. Artificial Intelligence, 1981, 17(1-3), pp. 185-203.

[99] Black, M.J. and Anandan, P. The robust estimation of multiple motions: Parametric and piecewise-smooth flow fields. Computer Vision and Image Understanding, 1996, 63(1), pp. 75-104.

[100] Zach, C., Pock, T., and Bischof, H. A duality based approach for realtime TV-L 1 optical flow. Pattern Recognition, 2007, pp. 214-223.

[101] Menze, M. and Geiger, A. Object scene flow for autonomous vehicles. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2015, pp. 3061-3070.

[102] Giebel, I., Gavrilu, D., and Schnörr, C. A Bayesian Framework for multi-cue 3d object tracking. Computer Vision(ECCV 2004), 2004, pp. 241-252.

[103] Breitenstein, M.D., Reichlin, F., Leibe, B., Koller-Meier, E., and Van Gool, L. Online multiperson tracking-by-detection from a single, uncalibrated camera. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2011, 33(9), pp.1820-1833.

[104] Andriyenko, A. and Schindler, K. Multi-target tracking by continuous energy minimization. In Proceedings of 2011 IEEE Conference on Computer Vision and Pattern Recognition(CVPR 2011), 2011, pp. 1265-1272.

[105] Xiang, Y., Alahi, A., and Savarese, S. Learning to track: Online multi-object tracking by decision making. In Proceedings of the IEEE International Conference on Computer Vision, 2015, pp. 4705-4713.

[106] Boser, B.E., Guyon, I.M., and Vapnik, V.N. A training algorithm for optimal margin classifiers. In Proceedings of the Fifth Annual Workshop on Computational Learning Theory, 1992, pp. 144-152.

[107] Paden, B., Cap, M., Yong, S.Z., Yershow, D., and Frazzolo, E. A survey of motion planning and control techniques for self-driving urban vehicles. IEEE Transactions on Intelligent Vehicles, 2016, 1(1) pp. 33-55.

[108] Gindele, T., Brechtel, S., and Dillmann, R. A probabilistic model for estimating driver behaviors and vehicle trajectories in traffic environments. In Proceedings of the 13th International IEEE Conference on Intelligent



Transportation Systems(ITSC), 2010, pp. 1625-1631.

[109] Aoude, G.S., Desaraju, V.R., Stephens, L.H., and How, J.P. Driver behavior classification at intersections and validation on large naturalistic data set. *IEEE Transactions on Intelligent Transportation Systems*, 2012, 13(2), pp. 724-736.

[110] Lefevre, S., Gao, Y., Vasquez, D., Tseng, H.E., Bajcsy, R., and Borrelli, F. Lane keeping assistance with learning-based driver model and model predictive control. In *Proceedings of the 12th International Symposium on Advanced Vehicle Control*, 2014, pp. 1-8.

[111] Gadepally, V., Krishnamurthy, A., and Ozguner, U. A framework for estimating driver decisions near intersections. *IEEE Transaction on Intelligent Transportation Systems*, 2014, 15(2), pp. 637-646.

[112] Gadepally, V., Krishnamurthy, A., and Ozguner, U. A Framework for Estimating Long Term Driver Behavior. *Journal of Advanced Transportation*, 2017, pp. 1-12.

[113] Bonnin, S., Weisswange, T.H., Kummert, F., and Schmuедderich, J. General behavior prediction by a combination of scenario-specific models. *IEEE Transactions on Intelligent Transportation Systems*, 2014, 15(4), pp. 1478-1488.

[114] Kumar, P., Perrollaz, M., Lefevre, S., and Laugier, C. Learning-based approach for online lane change intention prediction. In *Proceedings of the IEEE Intelligent Vehicles Symposium(IV 2013)*, 2013, pp. 797-802.

[115] 가동호, 이동훈 and 여화수. 보행자 특성을 고려한 예측형 보행자 충돌 경고 서비스 개발. *한국ITS학회 논문지*, 2019, 18(3), pp. 68-83.

[116] Hsu, C.W., Chang, C.C., and Lin, C.J. *A Practical Guide to Support Vector Classification*. Department of Computer Science, National Taiwan University, 2003, pp. 1-16.

[117] Krizhevsky, A., Sutskever, I., and Hinton, G.E. Imagenet classification with deep convolutional neural networks. In *Proceedings of Advances in Neural Information Processing Systems(NIPS 2012)*, 2012, pp. 1097- 1105.

[118] Medsker, L.R. and Jain, L.C. *Recurrent neural networks Design and Applications*. CRC Press, 2001, pp. 1- 389.

[119] Sak, H., Senior, A., and Beaufays, F. Long short-term memory recurrent neural network architectures for large scale acoustic modeling. In *Fifteenth Annual Conference of the International Speech Communication Association*, 2014, pp. 338-342.



[120] Janet, J.A.; Luo, R.C.; Kay, M.G. The essential visibility graph: An approach to global motion planning for autonomous mobile robots. In Proceedings of 1995 IEEE International Conference on Robotics and Automation, 1995, 2, pp. 1958-1963.

[121] Majeed, A. and Hwang, S.O. Path Planning Method for UAVs Based on Constrained Polygonal Space and an Extremely Sparse Waypoint Graph. Applied Sciences, 2021, 11, pp. 1-26.

[122] Gbadamosi, O.A. and Aremu, D.R. Design of a Modified Dijkstra's Algorithm for finding alternate routes for shortest-path problems with huge costs. In Proceedings of the 2020 International Conference in Mathematics, Computer Engineering and Computer Science (ICMCECS), 2020, pp. 1-6.

[123] Qing, G., Zheng, Z., and Yue, X. Path-planning of automated guided vehicle based on improved Dijkstra algorithm. In Proceedings of the 2017 29th Chinese control and decision conference (CCDC), 2017, pp. 7138-7143.

[124] Syed Abdullah, F., Iyal, S., Makhtar, M., and Jamal, A.A. Robotic Indoor Path Planning Using Dijkstra's Algorithm with Multi-Layer Dictionaries. In Proceedings of the 2015 2nd International Conference on Information Science and Security (ICISS), 2015, pp. 1-4.

[125] Kang, H., Lee, B.h., and Kim, K. Path Planning Algorithm Using the Particle Swarm Optimization and the Improved Dijkstra Algorithm. In Proceedings of the 2008 IEEE Pacific-Asia Workshop on Computational Intelligence and Industrial Application, 2009, 18, pp. 1002-1004.

[126] Fuhao, Z. and Jiping, L. An Algorithm of Shortest Path Based on Dijkstra for Huge Data. In Proceedings of the 2009 Sixth International Conference on Fuzzy Systems and Knowledge Discovery, 2009, 4, pp. 244-247.

[127] Fu, M., Li, J., and Deng, Z. A practical route planning algorithm for vehicle navigation system. In Proceedings of the Fifth World Congress on Intelligent Control and Automation, 2004, 6, pp. 5326-5329.

[128] Silva, M.C., Garcia, A.C.B., and Conci, A. A Multi-agent System for Dynamic Path Planning. In Proceedings of the 2010 Second Brazilian Workshop on Social Simulation, 2010, pp. 47-51.

[129] A* Algorithm. <http://web.mit.edu/eranki/www/tutorials/search/>.

[130] Baker, Z.K. and Gokhale, M. On the Acceleration of Shortest Path Calculations in Transportation Networks. In Proceedings of the 15th Annual IEEE Symposium on Field-Programmable Custom Computing Machines (FCCM 2007),



2007, pp. 23-34.

[131] Seo, W., Ok, S., Ahn, J., Kang, S., and Moon, B. An Efficient Hardware Architecture of the A-star Algorithm for the Shortest Path Search Engine. In Proceedings of the 2009 Fifth International Joint Conference on INC, IMS and IDC, 2009, pp. 1499-1502.

[132] Yanrong, H. and Yang, S.X. A knowledge based genetic algorithm for path planning of a mobile robot. In Proceedings of the IEEE International Conference on Robotics and Automation, 2004, 5, pp. 4350-4355.

[133] Stentz, A. Optimal and Efficient Path Planning for Unknown and Dynamic Environments; International Journal of Robotics and Automation, 2003, 10(3), pp. 285-287.

[134] Nilsson, N.J. Principles of Artificial Intelligence. Morgan Kaufmann, 1980.

[135] Stentz, A. The D* Algorithm for Real-Time Planning of Optimal Traverses. Technical Report, Robotics Institute, Carnegie Mellon University, 1994.

[136] Koenig, S. and Likhachev, M. D*Lite. In Eighteenth National Conference on Artificial Intelligence, 2002, pp. 476-483.

[137] Wang, J., Garratt, M.A., and Anavatti, S.G. Dynamic path planning algorithm for autonomous vehicles in cluttered environments. In Proceedings of the 2016 IEEE International Conference on Mechatronics and Automation, 2016, pp. 1006-1011.

[138] Ganapathy, V., Yun, S.C., and Chien, T.W. Enhanced D * Lite Algorithm for Autonomous Mobile Robot Velappa Ganapathy. International Journal of Applied Science and Technology, 1(1), pp. 58-73.

[139] Singh, S., Simmons, R., Smith, T., Stentz, A., Verma, V., Yahja, A., and Schwehr, K. Recent progress in local and global traversability for planetary rovers. In Proceedings of the 2000 IEEE International Conference on Robotics and Automation. Symposia(ICRA 2000), 2000, 2, pp. 1194-1200.

[140] LaValle, S.M. Rapidly-Exploring Random Trees: A New Tool for Path Planning. The annual research report, 1998.

[141] LaValle, S.M. Planning Algorithms. Cambridge University Press, 2006.

[142] LaValle, S.M., Kuffner, J.J. Jr. Randomized kinodynamic planning. International Journal of Robotics Research, 2001, 20(5), pp. 378-400.

[143] Huang, L. Rapidly Exploring Random Tree (RRT) and RRT*. 2018.

<http://wolfram.com/knowledgebase/source-information>

[144] Souissi, O., Benatitallah, R., Duvivier, D., Artiba, A., Belanger, N., and



Feyzeau, P. Path planning: A 2013 survey. In Proceedings of the 2013 International Conference on Industrial Engineering and Systems Management (IESM), 2013, pp. 1-8.

[145] Burchardt, H., Salomon, R. Implementation of Path Planning using Genetic Algorithms on Mobile Robots. In Proceedings of the 2006 IEEE International Conference on Evolutionary Computation, 2006, pp. 1831-1836.

[146] Elshamli, A., Abdullah, H.A., and Areibi, S. Genetic algorithm for dynamic path planning. In Proceedings of the Canadian Conference on Electrical and Computer Engineering, 2004, 2, pp. 677-680.

[147] Samadi, M. and Othman, M.F. Global Path Planning for Autonomous Mobile Robot Using Genetic Algorithm. In Proceedings of the 2013 International Conference on Signal-Image Technology Internet-Based Systems, 2013, pp. 726-730.

[148] L-Taharwa, I.A., Sheta, A., and Al-Weshah, M. A Mobile Robot Path Planning Using Genetic Algorithm in Static Environment. Journal of Computer Science, 2008, 4(4), pp. 341-344.

[149] Thomaz, C., Pacheco, M., and Vellasco, M. Mobile robot path planning using genetic algorithms. In International Work-Conference on Artificial Neural Networks, 2006, 1606, pp. 671-679.

[150] Yu, K., Lee, M., and Chi, S. Dynamic path planning based on adaptable Ant colony optimization algorithm. In Proceedings of the 2017 Sixth International Conference on Future Generation Communication Technologies (FGCT), 2017, pp. 1-7.

[151] Meng, L., Lin, C., Huang, H., and Cai, X. Vehicle routing plan based on ant colony and insert heuristic algorithm. In Proceedings of the 2016 35th Chinese Control Conference (CCC), 2016, pp. 2658-2662.

[152] Zhou, Z., Nie, Y., and Min, G. Enhanced Ant Colony Optimization Algorithm for Global Path Planning of Mobile Robots. In Proceedings of the 2013 International Conference on Computational and Information Sciences, 2013, pp. 698-701.

[153] Wang, Y., Lu, X., and Zuo, Z. Autonomous Vehicles Path Planning With Enhanced Ant Colony optimization. In Proceedings of the 2019 Chinese Control Conference (CCC), 2019, pp. 6633-6638.

[154] Wang, H., Guo, F., Yao, H., He, S., and Xu, X. Collision Avoidance Planning Method of USV Based on Improved Ant Colony Optimization Algorithm.



IEEE Access 2019, 7, pp. 52964–52975.

[155] Huadong, Z., Chaofan, L., and Nan, J. A Path Planning Method of Robot Arm Obstacle Avoidance Based on Dynamic Recursive Ant Colony Algorithm. In Proceedings of the 2019 IEEE International Conference on Power, Intelligent Computing and Systems (ICPICS), 2019, pp. 549–552.

[156] He, J., Wang, H., Liu, C., and Yu, D. UUV Path Planning for Collision Avoidance Based on Ant Colony Algorithm. In Proceedings of the 2020 39th Chinese Control Conference (CCC), 2020, pp. 5528–5533.

[157] Juntao, L., Tingting, D., Yuanyuan, L., and Yan, H. Study on Robot Path Collision Avoidance Planning Based on the Improved Ant Colony Algorithm. In Proceedings of the 2016 8th International Conference on Intelligent Human-Machine Systems and Cybernetics (IHMSC), 2016, 2, pp. 540–544.

[158] Qiannan, Z., Ziyang, Z., Chen, G., and Ruyi, D. Path planning of UAVs formation based on improved ant colony optimization algorithm. In Proceedings of the 2014 IEEE Chinese Guidance, Navigation and Control Conference, 2014, pp. 1549–1552.

[159] Chen, M. Toward Adaptive Ant Colony Algorithm. In Proceedings of the 2010 International Conference on Measuring Technology and Mechatronics Automation, 2010, 3, pp. 1035–1038.

[160] Wang, R., Zheng, D., Zhang, H., Mao, J., and Guo, N. A solution for simultaneous adaptive ant colony algorithm to memory demand vehicle routing problem with pickups. In Proceedings of the 2016 Chinese Control and Decision Conference (CCDC), 2016, pp. 2172–2176.

[161] Havlak, F. and Campbell, M. Discrete and continuous, probabilistic anticipation for autonomous robots in urban environments. IEEE Transactions on Robotics, 2013, 30(2), pp. 461–474.

[162] Ulbrich, S. and Maurer, M. Probabilistic online pomdp decision making for lane changes in fully automated driving. In Proceedings of 16th international IEEE conference on intelligent transportation systems(ITSC 2013), 2013, pp. 2063–2067.

[163] Brechtel, S., Gindele, T., and Dillmann, R. Probabilistic decision-making under uncertainty for autonomous driving using continuous pomdps. In Proceedings of 17th international IEEE conference on intelligent transportation systems(ITSC), 2014, pp. 392–399.

[164] Galceran, E., Cunningham, A.G., Eustice, R.M., and Olson, E. Multipolicy



decision-making for autonomous driving via changepoint-based behavior prediction: Theory and experiment. *Autonomous Robots*, 2017, 41(6), pp. 1367–1382.

[165] Wray, K. H., Witwicki, S. J., and Zilberstein, S. Online decision-making for scalable autonomous systems. In *International joint conference on artificial intelligence*, 2017, pp. 4768–4774.

[166] Katrakazas, C., Quddus, M., Chen, W.H., and Deka, L. Real-time motion planning methods for autonomous on-road driving : State-of-the-art and future research directions Elsevier *Transportation Research Part C: Emerging Technologies*, 2015, 60, pp. 416–442.

[167] Gu, T., Snider, J., M. Dolan, J. and Lee, J. Focused trajectory planning for autonomous on-road driving. *IEEE Intelligent Vehicles Symposium(IV)*, 2013, pp. 547–552.

[168] Paden, B., Cap, M., Yong, S.Z., Yershow, D., and Frazzolo, E. A survey of motion planning and control techniques for self-driving urban vehicles. *IEEE Transactions on Intelligent Vehicles*, 2016, 1(1), pp. 33–55.

[169] Mcnaughton, M. Parallel algorithms for real-time motion planning. *Doctoral Dissertation. Robotics Institute, Carnegie Mellon University*, 2011.

[170] González, D., Pérez, J., Milanés, V., and Nashashibi, F. A review of motion planning techniques for automated vehicles. *IEEE Transactions on Intelligent Transportation Systems*, 2015, 17(4), pp. 1135–1145.

[171] Paden, B., Čáp, M., Yong, S. Z., Yershov, D., and Frazzoli, E. A survey of motion planning and control techniques for self-driving urban vehicles. *IEEE Transactions on Intelligent Vehicles*, 2016, 1(1), pp. 33–55.

[172] Pivtoraiko, M., Knepper, R. A., and Kelly, A. Differentially constrained mobile robot motion planning in state lattices. *Journal of Field Robotics*, 2009, 26(3), pp. 308–333.

[173] McNaughton, M., Urmson, C., Dolan, J. M., and Lee, J.-W. Motion planning for autonomous driving with a conformal spatiotemporal lattice. In *2011 IEEE international conference on robotics and automation*, 2011, pp. 4889–4895.

[174] Xu, W., Wei, J., Dolan, J. M., Zhao, H., and Zha, H. A real-time motion planner with trajectory optimization for autonomous vehicles. In *2012 IEEE international conference on robotics and automation*, 2012, pp. 2061–2067.

[175] Gu, T., Dolan, J. M., and Lee, J.-W. Automated tactical maneuver discovery, reasoning and trajectory planning for autonomous driving. In *2016*



IEEE/RSJ international conference on intelligent robots and systems(IROS), 2016, pp. 5474-5480.

[176] Li, X., Sun, Z., Cao, D., He, Z., and Zhu, Q. Real-time trajectory planning for autonomous urban driving: Framework, algorithms, and verifications. IEEE/ASME Transactions on Mechatronics, 2015, 21(2), pp. 740-753.

[179] Gu, T., Atwood, J., Dong, C., Dolan, J. M., and Lee, J.-W. Tunable and stable real-time trajectory planning for urban autonomous driving. In 2015 IEEE/RSJ international conference on intelligent robots and systems(IROS), 2015, pp. 250-256.

[178] Fassbender, D., Heinrich, B.C., and Wuensche, H.-J. Motion planning for autonomous vehicles in highly constrained urban environments. In 2016 IEEE/RSJ international conference on intelligent robots and systems(IROS), 2016, pp. 4708-4713.

[179] Radaelli, R.R., Badue, C., Gonçalves, M.A., Oliveira-Santos, T., and De Souza, A.F. A motion planner for car-like robots based on rapidly-exploring random trees. In Ibero-American conference on artificial intelligence, 2014, pp. 469-480.

[180] Du, M., Mei, T., Liang, H., Chen, J., Huang, R., and Zhao, P. Drivers' visual behavior-guided rrt motion planner for autonomous on-road driving. Sensors, 2016, 16(1), pp. 1-19.

[181] Alia, C., Gilles, T., Reine, T., and Ali, C. Local trajectory planning and tracking of autonomous vehicles, using clothoid tentacles method. In Proceedings of 2015 IEEE intelligent vehicles symposium (IV), 2015, pp. 674-679.

[182] Mouhagir, H., Talj, R., Cherfaoui, V., Aioun, F., and Guillemard, F. Integrating safety distances with trajectory planning by modifying the occupancy grid for autonomous vehicle navigation. In Proceedings of 2016 IEEE 19th international conference on intelligent transportation systems(ITSC), 2016, pp. 1114-1119.

[183] Mouhagir, H., Cherfaoui, V., Talj, R., Aioun, F., and Guillemard, F. Trajectory planning for autonomous vehicle in uncertain environment using evidential grid. IFAC-PapersOnLine, 2017, 50(1), pp. 12545-12550.

[184] Howard, T.M. and Kelly, A. Optimal rough terrain trajectory generation for wheeled mobile robots. International Journal of Robotics Research, 2007, 26(2), pp. 141-166.

[185] Ferguson, D., Howard, T.M., and Likhachev, M. Motion planning in urban



environments. *Journal of Field Robotics*, 2008, 25(11-12), pp. 939-960.

[186] Li, X., Sun, Z., Cao, D., Liu, D., and He, H. Development of a new integrated local trajectory planning and tracking control framework for autonomous ground vehicles. *Mechanical Systems and Signal Processing*, 2017, 87, pp. 118-137.

[187] Cardoso, V., Oliveira, J., Teixeira, T., Badue, C., Mutz, F., Oliveira-Santos, T., Veronese, L., and De Souza, A.F. A model-predictive motion planner for the iara autonomous car. In *Proceedings of 2017 IEEE international conference on robotics and automation(ICRA)*, 2017, pp. 225-230.

[188] Poole, D.L. and Mackworth, A.K. *Artificial Intelligence: Foundations of Computational Agents*. Cambridge University Press, 2010.

[189] Medium. Robotic Path Planning: RRT and RRT*. <https://medium.com/@theclassytim/robotic-planning-rrt-and-rrt-212319121378>

[190] Funke, J., Theodosis, P., Hindiyeh, R., Stanek, G., Kritatakirana, K., Gerdes, C., Langer, D., Hernandez, M., Müller-Bessler, B., and Huhnke, B. Up to the limits: Autonomous audi tts. In *Proceedings of 2012 IEEE intelligent vehicles symposium*, 2012, pp. 541-547.

[191] Aström, K.J., and Murray, R.M. *Feedback systems: an introduction for scientists and engineers*. Princeton university press, 2010.

[192] Marino, R. , Scalzi, S. , and Netto, M. Nested PID steering control for lane keeping in autonomous vehicles. *Control Engineering Practice*, 2011, 19(12), pp. 1459-1467.

[193] Guidolini, R., De Souza, A.F., Mutz, F., and Badue, C. Neural-based model predictive control for tackling steering delays of autonomous cars. In *Proceedings 2017 International Joint Conference on Neural Networks(IJCNN)*, 2017, pp. 4324-4331.

[194] Zhao, P., Chen, J., Song, Y., Tao, X., Xu, T., and Mei, T. Design of a control system for an autonomous vehicle based on adaptive-pid. *International Journal of Advanced Robotic Systems*, 2012, 9(2), 44.

[195] Levinson, J., Askeland, J., Becker, J., Dolson, J., Held, D., Kammel, S., Kolter, J. Z., Langer, D., Pink, O., and Pratt, V. Towards fully autonomous driving: Systems and algorithms. In *Proceedings of 2011 IEEE intelligent vehicles symposium (IV)*, 2011, pp. 163-168.

[196] Garcia, C.E., Prett, D.M., and Morari, M. Model predictive control: theory and practice - a survey. *Automatica*, 1989, 25(3), pp.335-348.



- [197] Bemporad, A. Model Predictive Control. <http://cse.lab.imtlucca.it/~bemporad/teaching/ac/pdf/AC2-10-MPC.pdf>
- [198] Hui, J. (2018). Lane keeping in autonomous driving with Model Predictive Control & PID. https://medium.com/@jonathan_hui/lane-keepingin-autonomous-driving-with-model-predictive-control-50f06e989bc9
- [199] Samson, C. Control of chained systems application to path following and timevarying point-stabilization of mobile robots. IEEE Transactions on Automatic Control, 1995, 40(1), pp. 64-77.
- [200] Snider, J.M. Automatic Steering Methods for Autonomous Automobile Path Tracking, Robotics Institute, Carnegie Mellon University, 2009, pp. 1-78.
- [201] Koga, A., Okuda, H., Tazaki, Y., Suzuki, T., Haraguchi, K., and Kang, Z. Realization of different driving characteristics for autonomous vehicle by using model predictive control. In Proceedings of 2016 IEEE intelligent vehicles symposium (IV), 2016, pp. 722-728.
- [202] Kritayakirana, K., and Gerdes, J.C. Autonomous vehicle control at the limits of handling. Ph.D. thesis, Stanford University, 2012.
- [203] Brown, M., Funke, J., Erlien, S., and Gerdes, J.C. Safe driving envelopes for path tracking in autonomous vehicles. Control Engineering Practice, 2017, 61, pp. 307-316.
- [204] Laurence, V.A., Goh, J.Y., and Gerdes, J.C. Path-tracking for autonomous vehicles at the limit of friction. In Proceedings of 2017 American control conference(ACC), 2017, pp. 5586-5591.
- [205] ETSI, Etsi - ts 122 185 Requirements for V2X services. https://www.etsi.org/deliver/etsi_ts/122100_122199/122185/14.03.00_60/ts_122185v140300p.pdf.
- [206] Abboud, K., Omar, H.A., and Zhuang, W. Interworking of DSRC and cellular network technologies for V2X communications: A survey. IEEE transactions on vehicular technology, 2016, 65(12), pp. 9457- 9470.
- [207] Bouk, S.H., Kim, G., Ahmed, S.H., and Kim, D. Hybrid Adaptive Beaconing in Vehicular Ad Hoc Networks: A Survey. International Journal of Distributed Sensor Networks, 2015, 11(5), pp.1-16.
- [208] Festag, A. Cooperative intelligent transport systems standards in europe. IEEE Communications Magazine, 2014, 52(12), pp. 166-172.
- [209] IEEE, IEEE standard for wireless access in vehicular environments



(wave)-networking services. IEEE 1609 Working Group and others, 2016.

[210] ETSI. Ts 102636-5-1v1.2. Intelligent transport systems (ITS): vehicular communications; geonetworking: part 5: Transport protocols; sub-part 1 : Basic transport protocol.

[211] ETSI. Ts 102 636-4-1v1.2. Intelligent transport systems (ITS): vehicular communications; geonetworking: part 4 : Geographical addressing and forwarding for point-to-point and point-to-multipoint communications; sub-part 1 : Media-independent functionality.

[212] SAE. J2735: Dedicated short range communications (dsrc) message set dictionary.

https://www.etsi.org/deliver/etsi_en/302600_302699/3026360401/01.02.00_20/en_3026360401v010200a.pdf.

[213] Chevreuril, M. IVHW: an inte π ehicle hazard warning system concept within the DEUFRAKO program. e-Safety Congress and Exhibition, 2002. The Transportation Research Board.

[214] Franz, W., Hartenstein, H., and Mauve, M. InterVehicle Communications Based on Ad Hoc Networking Principles: The FleetNet Project. Universitätsverlag Karlsruhe, 2005.

[215] Hartenstein, H., Bochow, B., Ebner, A. et al. Position-aware ad hoc wireless networks for intervehicle communications: the Fleetnet project. In Proceedings of the 2nd ACM International Symposium on Mobile Ad Hoc Networking & Computing, 2001, pp. 259-262.

[216] Festag, A , Fu ß ler, H., Hartenstein, H. et al. Fleetnet: bringing car-to-car communication into the real world. Proceedings of the 11th World Congress on ITS, 2004, pp. 1-8.

[217] Reichardt, D., Miglietta, M., Moretti, L. et al. CarTALK 2000 : safe and comfortable driving based upon inter-vehicle-communication. In Proceedings of 2002 IEEE Intelligent Vehicle Symposium, 2002, 2, pp. 545-550.

[218] Morsink, P.L.J., Hallouzi, R., Dagli, I. et al. CarTALK 2000 : development of a co-operative ADAS based on vehicle-to-vehicle communication. In Proceedings of 10th World Congress and Exhibition on Intelligent Transport Systems and Services, 2003.

[219] Yi, C., Gehlen, G., Jodlauk, G. et al. A flexible application layer protocol for automotive communications in cellular networks. In Proceedings of 15th World Congress on Intelligent Transportation Systems(ITS 2008), 2008, pp. 1-9.



[220] Schulze, M., Nocker, G., and Bohm, K. PREVENT: a European program to improve active safety. In Proceedings of 5th International Conference on Intelligent Transportation Systems Telecommunications, 2005, pp. 1-5.

[230] Hiller, A., Hinsberger, A., Strassberger, M., and Verburg, D. Results from the WILLWARN project. In Proceedings of 6th European Congress and Exhibition on Intelligent Transportation Systems and Services, 2007, pp. 1-8.

[231] Festag, A., Noecker, G., Strassberger, M. et al. NoW - Network on wheels: Project objectives, technology and achievements. In Proceedings of 5th International Workshop on Intelligent Transportation (WIT), 2008, pp. 211-216.

[232] Toulminet, G., Boussuge, J., and Lurgeau, C. Comparative synthesis of the 3 main European projects dealing with Cooperative Systems (CVIS, SAFESPOT and COOPERS) and description of COOPERS Demonstration Site 4. In Proceedings of 2008 11th International IEEE Conference on Intelligent Transportation Systems(ITSC 2008), 2008, pp. 809-814.

[233] Bonnefoi, F., Bellotti, F., Scendzielorz, T., and Visintainer, F. SAFESPOT applications for infrastructure based co-operative road safety. In Proceedings of 14th World Congress and Exhibition on Intelligent Transport Systems and Services, 2007, pp. 1-8.

[234] Stubing, H., Bechler, M., Heussner, D. et al. simTD: a carto-X system architecture for field operational tests [topics in automotive networking]. IEEE Communications Magazine, 2010, 48(5), pp. 148-154.

[235] Papathanassiou, A. and Khoryaev, A. Cellular V2X as the essential enabler of superior global connected transportation services. IEEE 5G Tech Focus, 2017, 1(2), pp. 1-2.

[236] Choi, E.-H. Crash factors in intersection- related crashes: An onscene perspective. Technical report, 2010.

[237] ROSPA. Road safety factsheet-overtaking.
<https://www.rospa.com/media/documents/road-safety/factsheets/overtaking-factsheet.pdf>

[238] Hobert, L., Festag, A., Llatser, I. et al. Enhancements of V2X communication in support of cooperative autonomous driving. IEEE Communications Magazine, 2015, 53(12), pp. 64-70.

[239] De La Fortelle, A., Qian, X., Diemer, S. et al. Network of automated vehicles: the autonet 2030 vision. ITS World Congress, 2014, pp. 1-9.

[240] Marjovi, A., Vasic, M., Lemaitre, J., and Martinoli, A. Distributed



graph-based convoy control for networked intelligent vehicles. In Proceedings of 2015 IEEE Intelligent Vehicles Symposium (IV), 2015, pp. 138-143.

[241] Khan, U. , Basaras, P., Schmidt-Thieme, L. et al. Analyzing cooperative lane change models for connected vehicles. In Proceedings of 2014 International Conference on Connected Vehicles and Expo (ICCVE). 2014, pp. 565- 570.

[242] Kesting, A., Treiber, M., and Helbing, D. General Lane-Changing Model MOBIL for Car-Following Models. Transportation Research Record, 2007, 1999(1), pp. 86-94.

[243] Zheng, K. and Zheng, Q. Chatzimisios, P. et al. Heterogeneous vehicular networking : a survey on architecture, challenges, and solutions. IEEE Communications Surveys and Tutorials, 2015, 17(4), pp. 2377- 2396.

[244] Kenney, J. B. Dedicated short-range communications (DSRC) standards in the United States. Proceedings of the IEEE, 2011, 99(7), pp. 1162-1182.

[245] Zheng, K., Zheng, Q., Yang, H. et al. Reliable and efficient autonomous driving : the need for heterogeneous vehicular networks. IEEE Communications Magazine, 2015, 53(12), pp. 72-79.

[246] Isento, J. N.G. , Rodrigues, J.J.P.C. , Dias, J.A. F. F. et al, (2013). Vehicular delay-tolerant networks? A novel solution for vehicular communications. IEEE Intelligent Transportation Systems Magazine 5 (4): 10-19.

[247] Dias, J.A. F.F., Rodrigues, J.J.P.C., Kumar, N., and Saleem, K. Cooperation strategies for vehicular delay-tolerant networks. IEEE Communications Magazine, 2015, 53(12), pp. 88-94.

[248] Ghosh, A., Paranthaman, V.V., Mapp, G. et al. Enabling seamless V2I communications: toward developing cooperative automotive applications in VANET systems. IEEE Communications Magazine, 2015, 53(12), pp. 80-86.

[249] NHTSA Federal motor vehicle safety standards; V2V communications. Federal Register, 2017, 82(8), pp.3854-4019.

[250] Joan Lowy, Apnewsbreak: Gov' t won' t pursue talking car mandate, <https://apnews.com/article/9a605019eeba4ad2934741091105de42>

[251] US-DOT. ITS standards program development activities International harmonization.

<https://www.standards.its.dot.gov/DevelopmentActivities/IntlHarmonization>

[252] Jo, K., Kim, C., Sunwoo, M. Simultaneous Localization and Map Change Update for the High Definition Map-Based Autonomous Driving Car. Sensors. 2018, 18(9), pp. 1-16.



[253] Department of Economic and Social Affairs (DESA), Population Division, “The 2017 revision, key findings and advance tables,” in World Population Prospects. United Nations, 2017, no. ESA/P/WP/248.

[254] Deloitte. 2019 deloitte global automotive consumer study – advanced vehicle technologies and multimodal transportation, global focus countries. <https://www2.deloitte.com/content/dam/Deloitte/us/Documents/manufacturing/us-global-automotive-consumer-study-2019.pdf>. [Retrieved May 19, 2019].

[255] Federation Internationale de l’Automobile (FiA) Region 1. The automotive digital transformation and the economic impacts of existing data access models.

https://www.fiaregion1.com/wp-content/uploads/2019/03/The-Automotive-Digital-Transformation_Full-study.pdf

[256] Rajamani, R. Vehicle dynamics and control. Springer Science & Business Media, 2011.

[257] Hafner, M.R., Cunningham, D., Caminiti, L., and Del Vecchio, D. Cooperative collision avoidance at intersections: Algorithms and experiments. IEEE Transactions on Intelligent Transportation Systems, 2013, 14(3), pp. 1162–1175.

[258] Colombo, A. and Del Vecchio, D. Efficient algorithms for collision avoidance at intersections. In Proceedings of the 15th ACM international conference on Hybrid Systems: Computation and Control, 2012, pp. 145–154.

[259] Ross, P. E. The audi a8: the world’s first production car to achieve level 3 autonomy. IEEE Spectrum, 2017.

[260] Gold, C., Körber, M., Lechner, D., and Bengler, K. Taking over control from highly automated vehicles in complex traffic situations: the role of traffic density. Human factors, 2016, 58(4), pp. 642–652.

[261] Merat, N., Jamson, A. H., Lai, F. C., Daly, M., and Carsten, O. M. Transition to manual: Driver behaviour when resuming control from a highly automated vehicle. Transportation research part F: traffic psychology and behaviour, 2014, 27, pp. 274–282.

[262] Ackerman, E. Toyota’s gill pratt on self-driving cars and the reality of full autonomy. IEEE Spectrum, 2017.

[263] Lavrinc, D. This is how bad self-driving cars suck in rain.

<https://jalopnik.com/this-is-how-bad-self-driving-cars-suck-in-therain-1666268433>.

[264] Davies, A. Google’s self-driving car caused its first crash.



- <https://www.wired.com/2016/02/googles-self-driving-car-may-caused-first-crash/>.
- [265] McFarland, M. Who's responsible when an autonomous car crashes?
<https://money.cnn.com/2016/07/07/technology/tesla-liability-risk/index.html>
- [266] Lee, T. B. Autopilot was active when a tesla crashed into a truck, killing driver.
<https://arstechnica.com/cars/2019/05/feds-autopilot-was-active-during-deadly-march-tesla-crash/>
- [267] D' Onfro, J. 'I hate them' : Locals reportedly are frustrated with alphabet's self-driving cars.
<https://www.cnbc.com/2018/08/28/locals-reportedly-frustrated-with-alphabets-way-to-self-driving-cars.html>
- [268] Bonnefon, J.-F., Shariff, A., and Rahwan, I. The social dilemma of autonomous vehicles. *Science*, 2016, 352(6293), pp. 1573-1576.
- [269] Bonnefon, J., Shariff, A.F., and Rahwan, I. Autonomous vehicles need experimental ethics: Are we ready for utilitarian cars? 2015, aArXiv, abs/1510.03346.
- [270] Tian, Y., Pei, K., Jana, S., and Ray, B. Deeptest: Automated testing of deep-neural-network-driven autonomous cars. In *Proceedings of the 40th International Conference on Software Engineering*, 2018, pp. 303-314.



2022 _____ **ISSUE PAPER** _____

경남 창원시 의창구 창원대로 18번길 46

Tel. 055-259-3013 Fax. 055-259-3020

www.gntp.or.kr ☎