정적 장애물 회피를 위한 경로 계획: ADAM III

최 희 재 · 송 봉 섭*

아주대학교 기계공학과

Path Planning for Static Obstacle Avoidance: ADAM III

Heejae Choi · Bongsob Song*

Department of Mechanical Engineering, Ajou University, Gyeonggi 443-749, Korea
(Received 31 October 2013 / Revised 5 March 2014 / Accepted 12 March 2014)

Abstract : This paper presents a path planning algorithm of an autonomous vehicle (ADAM III) for collision avoidance in the presence of multiple obstacles. Under the requirements that a low-cost GPS is used and its computation should be completed with a sampling time of sub-second, heading angle estimation is proposed to improve performance degradation of its measurement and a hierarchical structure for path planning is used. Once it is decided that obstacle avoidance is necessary, the path planning consists in three steps: waypoint generation, trajectory candidate generation, and trajectory selection. While the waypoints and the corresponding trajectory candidates are generated based on position of obstacles, the final desired trajectory is determined with considerations of kinematic constraints as well as an optimal condition in a term of lateral deviation. Finally the proposed algorithm was validated experimentally through field tests and its demonstration was performed in Autonomous Vehicle Competition (AVC) 2013.

Key words : Autonomous vehicle(자율주행 자동차), Obstacle avoidance(장애물 회피), Path planning(경로 계획), Trajectory generation(궤적 생성)

1. 서 론

최근 무인자율주행차량에 대한 관심이 급격히 증가하고 있다. 대표적인 예로 구글사에서 개발한 무인자동차를 들 수 있으며 벤츠社에서는 2020년에 자율주행 자동차를 상용화하겠다는 목표를 발표한 바 있다.¹⁾ 해외의 경우 회사뿐만 아니라 대학 내에서도 자율주행 차량에 대한 연구가 꾸준히 이루어지고 있다. 미국에서는 2003년 DARPA에서 주관하는 Grand Challenge에 이어서 2007년에는 도심환경에서 주행하는 Urban Challenge까지 개최하여 기업과 대학이 협력하여 자율주행 차량 개발에 대한 연

구가 활발히 이루어지고 있다.^{2,3)}

국내에서는 2010년과 2012년에 각각 현대기아자동차 그룹에서 주최하는 자율주행 자동차 경진대회 (autonomous vehicle competition, AVC)가 열렸으며 2년을 주기로 지속적으로 개최되고 있다.^{4,5)} 2013년 10월에는 자동차 부품 연구원에서 주관하는 AVC가 열렸다. 이전 AVC와 차이점으로는 Fig. 1에서 보는 바와 같이 전방 라이다와 카메라, 그리고 GPS가 주어져 있고 추가적으로 라이다를 한 개 더 장착할 수 있는 제한조건에서 자율주행 자동차를 개발하도록 하였다. 이는 하드웨어가 비슷한 상황에서 알고리즘의 경쟁력을 평가하려는데 의의가 있었다. 더 나아가 AVC의 미션을 살펴보면 장애물 회피, 횡단보도 인지 및 정차, 표지판 인지, 교차로 주행 등이 있지만 본 논문에서는 단일 또는 복합 장애물 회피를 위한

*A part of this paper was presented at the KSAE 2013 Annual Conference and Exhibition

*Corresponding author, E-mail: bsong@ajou.ac.kr

Fig. 1 Test vehicle for static obstacle avoidance

경로 계획 알고리즘을 중점적으로 설명하고자 한다.

장애물 회피를 위한 경로 생성에 관련된 연구 동향을 살펴보면 크게 두 가지로 구분할 수 있다. 첫 번째로 포텐셜 필드를 사용하는 알고리즘과 같이 경유점 없이 궤적을 한 번에 생성하는 방법이 있으나 실차에 적용된 연구를 찾아보기는 거의 어렵다.⁶⁾ 두 번째로는 경유점을 선정한 후 궤적을 생성하는 방법이다. 예로는 A*와 D* 알고리즘, RRT (Rapidly-exploring Random Tree) 알고리즘, 보르노이(Voronoi) 다이어그램, 그리고 이산 최적화(discrete optimization)가 있으며 국내·외 자율주행 자동차 경진대회를 통해 장애물 회피 알고리즘이 검증되었다.^{2-5,10)}

위에서 언급된 대부분의 경우 고사양의 GPS를 사용하였지만 본 연구에서는 저가의 GPS를 사용하였을 경우 발생할 수 있는 문제점을 해결하고 단순한 장애물 시나리오와 복잡한 시나리오를 모두 고려하여 경로 계획을 하기 위한 방법에 대해서 제안하고자 한다. 즉, 장애물을 거리에 따라 계층을 나누고 이에 따라 동적 경유점을 생성하고, 주행 궤적 후보를 생성한다. 최종적으로 기구학적 제한조건을 만족하면서 횡방향 변화량이 적은 주행 궤적을 최종적으로 선정하는 경로 계획 알고리즘을 제안하고자 한다. 더 나아가 실차를 기반으로 실시간으로 경로 계획이 이루어지고 실제 장애물을 피할 수 있음을 실험적으로 검증하고자 한다.

본 논문의 구성은 우선 2장에서 하드웨어 및 소프트웨어에 대한 시스템 구성에 대해서 대략적으로 설명하며 3장에서는 경로 계획 알고리즘을 소개한다. 특히 주행모드 판단, 동적 경유점 생성, 주행 궤

적 후보군 생성, 궤적 선택에 대해서 순차적으로 설명하고자 한다. 마지막으로 4장에서는 실차기반 실험 결과를 보이하고자 한다.

2. 시스템 구성

2.1 하드웨어 구성

정적 장애물 회피를 위한 센서, 프로세서를 포함한 차량 하드웨어 구성도는 Fig. 1에서 보는 바와 같이 2대의 프로세서로 구성되어 있으며 인지 프로세서(perception processor)와 제어 프로세서(control processor)로 구분된다. 장애물 회피를 위한 센서의 구성은 라이다(lidar)와 GPS가 있으며 자세한 사양은 Table 1에 요약되어 있다.

Table 1 Hardware specifications of a test vehicle

Hardware	Manufacturer and model	Specifications
Perception processor	NI PXI-8110	2.26 GHz Quad-core
Control processor	Advantech UNO-2176	1.4 GHz Single-core
Lidar	SICK LMS 511	180 deg FOV with 0.5 deg angular resolution 80m maximum range
GPS	JY system B20	0.75 m RMS horizontal error 10 Hz output rate

2.2 소프트웨어 구성

제어 프로세서의 소프트웨어 구성은 Fig. 2에서 보는 바와 같이 계획기(planner)와 제어기(controller)로 구성되어 있으며 장애물에 대한 정보는 인지 프로세서로부터 CAN통신을 통하여 전달받는다. 인지 프로세서의 경우 내부적으로 라이다로부터 측정된 값에 대하여 probabilistic data association filter (PDAF)를 사용하여 장애물을 감지하고 추적하며 최종적으로 장애물에 대한 위치정보를 제어 프로세서로 전달하게 된다.⁷⁾

계획기의 경우 다시 주행모드 계획기(manuever planner), 경로 계획기(path planner), 위치(pose) 예측기로 나뉜다. 주행모드 계획기는 충돌 위험도에 따라 크게 일반 주행과 회피로 구분하여 해당 주행모드(manuever)를 제공하고, 경로 계획기에서는 경유

Fig. 2 Schematic function diagram of control processor for obstacle avoidance

점을 생성하고 주행 궤적을 생성한다. 또한, 위치 예측기의 경우 GPS를 통해 들어오는 위치 정보와 헤딩 각 정보를 필터링하여 좀 더 정확한 정보를 전달하는 역할을 담당한다. 마지막으로 제어기에 속해 있는 횡방향 제어기(lateral controller)에서는 주행 궤적을 추종하기 위한 목표 조향각을 생성한다.⁸⁾

3. 경로 계획 알고리즘

장애물 회피를 위한 경로 계획기는 Fig. 3에서 보는 바와 같이 경유점 생성, 주행 궤적 후보군 생성, 그리고 최종 주행 궤적 선택으로 구성되어 있다. 우선 경유점(waypoint)은 미리 주어진 경유점을 주행하는 정적(static) 경유점과 장애물 회피를 위하여 생성하는 동적(dynamic) 경유점으로 구분된다. 여기서 어떠한 종류의 경유점을 생성할 지는 Fig. 2의 주

Fig. 3 Schematic diagram of path planner

행모드 계획기의 주행모드에 의해서 결정된다. 즉 일반 주행모드와 장애물 회피 모드에 의해서 경유점의 종류가 결정된다. 따라서 본 절에서는 장애물 회피 모드가 결정되는 주행모드 판단에 대해서 먼저 간략히 설명을 하고 경로 계획기의 세 모듈에 대해서 순차적으로 설명하고자 한다.

3.1 주행 모드 판단

장애물 회피 모드를 판단하는 과정은 크게 두 가지로 구성이 되어 있다. 우선 경로 계획기에서 생성된 주행 궤적 내에 가까이 장애물이 위치되어 있는지는 판단하는 것과 해당하는 장애물에 대한 충돌 위험도를 판단하는 것으로 이루어져 있다.

주행 궤적 근처에 장애물이 위치되어 있는지를 판단하기 위해서 먼저 주행 궤적 선상에서 가장 가까운 장애물과의 최단거리와 안전거리(δ)를 다음과 같이 비교하여 장애물 회피의 필요성을 판단하였다.

$$\|p(u) \perp R_i\| \approx \|\overline{p(u_s)p(u_{s+1})} \perp R_i\| \leq \delta \quad (1)$$

여기서 $p(u)$ 는 주행 궤적을 뜻하며, $p(u_i)$ 는 주행궤적 상의 점을 의미한다. $\overline{p(u_s)p(u_{s+1})}$ 은 R_i 와 가장 가까운 주행 궤적 상의 두 점의 선분을 뜻하여 $\perp$ 는 점과 직선간의 최단거리를 뜻하고, 안전거리 $\delta = w_i + \alpha$, w_i 는 i 번째 장애물의 폭이며, α 은 차량 폭과 GPS의 오차범위를 고려한 설계 상수이다.

두 번째로 장애물 회피의 필요성이 판단된 후 장애물 회피를 위한 주행 궤적 생성 시점을 결정할 필요가 있다. 주행 궤적 생성 시점은 장애물과의 충돌 예상시간을 예측하여 결정하였으며 다음과 같이 정의되는 Time-To-Collision(TTC)를 이용하여 주행 경로 재생성 시점을 결정한다.

$$\text{Maneuver} = \begin{cases} \text{avoidance} & \text{if } TTC = \frac{R_w}{v} \leq \beta \\ \text{driving} & \text{otherwise} \end{cases} \quad (2)$$

여기서 v 는 차량속도, β 은 설계 상수이며, R_w 는 식 (1)을 만족하는 장애물에 대한 상대거리를 뜻한다.

마지막으로 Table 1에서 요약된 것과 같이 사용된 GPS는 위치 및 헤딩 각에 대한 측정 오차를 가지고 있어 정확한 주행모드 판단과 주행 궤적 생성을 위

Fig. 4 Comparison of trajectories based on estimation and measurement of GPS

해서는 위치와 헤딩 각에 대한 필터링이 필요하다. 예를 들어서 Fig. 4에서 보는 바와 같이 6개의 장애물이 놓여있는 시나리오의 경우 저가 GPS의 헤딩 각 측정값을 기반으로 주행 궤적을 생성하였다(Fig. 4의 점선 참조). 실제 장애물 사이로 차량이 통과할 수 있음에도 불구하고 장애물의 중심에서 안전거리(δ)를 반경으로 이룬 원과 교차됨을 확인할 수 있다. 이러한 경우 장애물 사이를 통과하기 보다는 우회하는 주행 궤적의 새롭게 생성될 수 있다.

이러한 문제점을 해결하기 위해서 비선형 관측기를 적용하여 위치와 헤딩 각에 대한 노이즈를 최소화하려고 하였다.⁹⁾ Fig. 4에서 보는 바와 같이 관측기를 기반으로 추정된 헤딩 각을 이용한 경우 장애물 사이를 통과할 수 있는 주행 궤적이 생성될 수 있음을 확인할 수 있다(Fig. 4의 검은 색 굵은 실선 참조).

3.2 동적 경유점 생성

동적 경유점을 생성하기 위하여 먼저 인지된 장애물의 거리에 따라 다음과 같이 계층을 분류하였다. 이는 정적 장애물이 놓여 있는 시나리오와 밀접하여 동적 경유점의 계층 정보를 기반으로 각기 다른 궤적 후보군을 생성하기 위함이다. 우선 동적 경유점의 거리에 따라 계층을 구분하기 위하여 다음과 같이 정의하였다.

$$R_k^j = \begin{cases} R_k^1 & \text{if } R_{\min}^1 \leq R_i \leq R_{\min}^1 + \gamma \\ R_k^2 & \text{if } R_{\min}^1 + \gamma \leq R_i \leq R_{\min}^2 + \gamma \\ R_k^3 & \text{otherwise} \end{cases} \quad (3)$$

여기서 위첨자 j 는 장애물의 계층을 뜻하며, k 의 최대 값을 각 계층별로 l, m, n 이라고 했을 경우 $l+m+n$

Fig. 5 Dynamic waypoint generation

은 인지된 장애물 총 개수(p)와 일치한다. 또한 γ 는 설계 상수이고 $R_{\min}^1$ 과 $R_{\min}^2$ 는 아래와 같이 정의된다.

$$R_{\min}^1 = \min_{1 \leq i \leq p} \|R_i\|, \\ R_{\min}^2 = \min_{1 \leq i \leq p} \|R_i\| \text{ subject to } R_i > R_{\min}^1 + \gamma$$

다음으로 동적 경유점의 생성은 Fig. 5에서 보는 바와 같이 동일 계층내의 인접해 있는 두 장애물이 존재하는 경우 두 장애물의 중심점을 이은 직선에서 각각의 장애물의 폭(w_k^j)을 빼고 그 중심에 동적 경유점을 아래와 같이 생성하였다.

$$DW_k^j = R_2^j + \frac{\|R_2^j - R_1^j\| - \left(\frac{w_2^j + w_1^j}{2}\right) + w_2^j}{2} \vec{u} \quad (4)$$

여기서 $\|R_2^j - R_1^j\| > 2\delta$ 를 만족하는 경우에만 생성을 하였으며 $\vec{u}$ 는 R_2^j 에서 R_1^j 방향으로의 단위벡터를 뜻한다.

장애물이 양 끝에 있는 경우와 계층 내에 장애물이 1 개인 경우에는 장애물을 기준으로 원의 접점을 이용하여 다음과 같이 생성한다(Fig. 5의 DW_1^j 참조).

$$DW_k^j = d_k^j \angle \theta_k^j \text{ for } k = 1 \text{ or } l, m, n \quad (5)$$

여기서

$$d_k^j = \sqrt{\|R_k^j\|^2 - \left(1.5\alpha + \frac{w_k^j}{2}\right)^2}, \theta_k^j = \tan^{-1} \left(\frac{3\alpha + w_k^j}{2d_k^j} \right) + \angle R_k^j$$

3.3 주행 궤적 후보 생성

식 (3)에 의해 계층이 구분되어 있는 동적 경유점을 기반으로 세 가지 경우로 나누어 주행 궤적 후보

를 생성한다. 즉, 첫 번째 계층에만 동적 경유점이 존재하는 경우, 첫 번째와 두 번째 계층에만 동적 경유점이 존재하는 경우, 마지막으로 모든 계층에 동적 경유점이 존재하는 경우로 나누어 주행 궤적 후보를 생성한다.

첫 번째 계층에만 ($l+1$)개의 동적 경유점이 존재하는 경우에는 Fig. 6(a)에서 보는 바와 같이 차량에서 동적 경유점까지 연결되는 곡선과 다시 동적 경유점에서 정적 경유점까지 연결되는 곡선으로 구성되어있다. cubic Bezier curve 방법을 이용하여 주행 궤적 후보를 생성하기 위해서는 총 7개의 궤적 제어점이 필요하다.⁸⁾ 궤적 제어점을 선정하는 방법은 다음과 같이 정의된다.

$$\begin{aligned} p_0 &= 0, p_3 = DW_k^j, p_6 = SW_k \\ p_1 &= p_0 + l_1 \angle \psi, p_2 = p_3 - l_1 \angle \phi_1 \\ p_4 &= p_3 + l_2 \angle \phi_1, p_5 = p_6 - l_1 \angle \phi_2 \end{aligned} \quad (6)$$

여기서 $p_i \in \mathbf{R}^2$, ψ 는 차량의 헤딩 각, SW_k 는 정적 경유점, l_1 과 l_2 는 다음과 같이 정의되며

$$l_1 = \epsilon \|p_0 - p_3\|, l_2 = \epsilon \|p_3 - p_6\| \quad (7)$$

이 때 $\epsilon = 0.5$ 를 사용하였고 ϕ_1 와 ϕ_2 는 다음과 같이 각각 정의된다.

$$\begin{aligned} \phi_1 &= \begin{cases} \theta_k^j & \text{for } k=1 \text{ or } l+1 \\ \angle(DW_k^j - SW_k) & \text{for } k=2, \dots, l \end{cases} \\ \phi_2 &= \angle(SW_{k+1}^j - SW_k) \end{aligned}$$

단, 위 방법으로 생성된 주행 궤적 후보 중에 대하여 식 (1)을 만족하는 경우는 주행 궤적 후보에서 제외된다.

다음으로 첫 번째와 두 번째 계층에만 동적 경유점이 ($l+1$)와 ($m+1$)개가 각각 존재하는 경우, 위에서 설명한 ($l+1$)개의 주행 궤적 후보를 모두 포함하고 식 (6)을 이용하여 다음과 같이 추가적으로 주행 궤적 제어점을 정의하여 주행 궤적 후보를 생성하였다.

$$p_2 = p_3 - l_{1r}^j \angle \phi_q^j, p_4 = p_3 + l_{2r}^j \angle \phi_q^j \quad (8)$$

여기서 $q=1$ 또는 2, ϕ_q^j 는 다음과 같이 정의된다.

$$\begin{aligned} \phi_1^j &= \angle(DW_{k_1}^j - p_3) \\ k_1^* &= \arg \min_{1 \leq k \leq l \text{ or } m} \|p_3 - DW_k^j\| \text{ for } t = \begin{cases} 2 & \text{if } j=1 \\ 1 & \text{otherwise} \end{cases} \end{aligned}$$

$$\begin{aligned} \phi_2^j &= \angle(DW_{k_2}^j - p_3) \\ k_2^* &= \arg \min_{\substack{1 \leq k \leq l \text{ or } m \\ k \neq k_1^*}} \|p_3 - DW_k^j\| \text{ for } t = \begin{cases} 2 & \text{if } j=1 \\ 1 & \text{otherwise} \end{cases} \end{aligned}$$

여기서 $DW_{k_1}^j$ 와 $DW_{k_2}^j$ 는 p_3 와 다른 계층에 있는 가장 가까운 동적 경유점과 두 번째로 가까운 점을 각각 뜻하며 차등적인 길이를 의미하는 l_{1r}^j 와 l_{2r}^j 는 다음과 같다.

$$\begin{aligned} l_{1r}^j &= \begin{cases} \epsilon_r \|p_3 - p_6\| & \text{if } j=3 \\ l_1 & \text{otherwise,} \end{cases} \\ l_{2r}^j &= \begin{cases} l_2 & \text{if } j=3 \\ \epsilon_r \|p_0 - p_3\| & \text{otherwise,} \end{cases} \\ \epsilon_r &= \{0.1, 0.2, \dots, 1\} \end{aligned} \quad (9)$$

예를 들어서 Fig. 6(b)에서 보는 바와 같이 p_3 가 DW_k^j 인 경우 식 (8)에 의거해서 p_3 와 가장 가까운 동적 경유점 $DW_{k_1}^j$ 와 이루는 각을 ϕ_1^j 로 정의하고 두 번째로 가까운 동적 경유점 $DW_{k_2}^j$ 와 이루는 각을 ϕ_2^j 로 정의한다. 다음으로 식 (9)의 ϵ_r 을 차등적으로 적용한 후 식 (1)을 만족하는 주행 궤적 후보군을 삭제하면 Fig. 6(b)에서 보는 바와 같이 총 11개의 주행 궤적 후보군이 생성된다.

마지막으로 모든 계층에 동적 경유점이 존재하는 경우 두 번째에서 설명한 주행 궤적 후보를 모두 포함하고 식 (8)의 ϕ_q^j 을 변경하여 다음과 같이 생성한다.

$$\begin{aligned} \phi_1^j &= \angle(DW_{k_1}^j - p_3) \\ k_1^* &= \arg \min_{1 \leq k \leq m \text{ or } n} \|p_3 - DW_k^j\| \text{ for } t = \begin{cases} 3 & \text{if } j=2 \\ 2 & \text{otherwise} \end{cases} \end{aligned}$$

$$\begin{aligned} \phi_2^j &= \angle(DW_{k_2}^j - p_3) \\ k_2^* &= \arg \min_{\substack{1 \leq k \leq m \text{ or } n \\ k \neq k_1^*}} \|p_3 - DW_k^j\| \text{ for } t = \begin{cases} 3 & \text{if } j=2 \\ 2 & \text{otherwise} \end{cases} \end{aligned}$$

Fig. 6(c)에서 보는 바와 같이 p_3 가 DW_k^j 인 경우 식 (8)에 의거해서 p_3 와 가장 가까운 동적 경유점 $DW_{k_1}^j$ 와 이루는 각을 ϕ_1^j 로 정의하고 두 번째로 가까운 동적 경유점 $DW_{k_2}^j$ 와 이루는 각을 ϕ_2^j 로 정의한다. 다음으로 식 (9)의 ϵ_r 을 차등적으로 적용한 후 식 (1)을 만족하는 주행 궤적 후보군을 삭제하면 Fig. 6(c)에서 보는 바와 같이 총 13개의 주행 궤적 후보군이 생성된다.

(a) Case 1: no dynamic waypoint in both 2nd or 3rd layer

(b) Case 2: dynamic waypoint in both 1st or 2nd layer

(c) Case 3: dynamic waypoint in all layers

Fig. 6 Trajectory candidates generation

3.4 주행 궤적 선택

위에서 생성한 주행 궤적 후보군 중 최적의 주행 궤적을 선택하기 위해서는 조향 액추에이터의 동역학적 제한조건을 이용하여 주행 궤적 후보를 제외하는 과정과 최적 조건을 이용하여 최종 주행 궤적을 선택하는 과정으로 구성되어 있다. 우선 조향 액추에이터의 시간지연과 같은 동특성을 고려하여 과거에 생성된 궤적의 진행방향과 새롭게 생성된 궤적 후보의 주행 예측 방향의 차이가 특정 값보다 큰 경우에는 주행 궤적 후보를 삭제하고자 한다. 주행

궤적의 진행방향은 현재 위치에서 일정 거리가 떨어진 점과의 각을 이용하여 아래와 같이 주행 궤적 후보 제외 조건을 정의하였다.

$$|\angle p_t(u_{s^*}) - \angle p_{old}(u_{i^*})| \geq \zeta \text{ for } t=1, \dots, v \quad (10)$$

여기서 $p_t(u)$ 는 t 번째 주행 궤적 후보를 뜻하며 v 는 주행 궤적 후보 수이다. 또한 $p_{old}(u)$ 는 재생성하기 전의 과거 주행 궤적을 뜻하며 ζ 는 설계 상수이며 s^* 와 i^* 는 아래와 같다.

$$s^* = \arg \min_s |\eta - \|p_t(u_s)\||$$

$$i^* = \arg \min_{i \geq j^*} |\eta - \|p_{old}(u_i) - p_{old}(u_{j^*})\||$$

이 때 j^* 는 주행 궤적 후보는 생성하는 위치에서 과거 주행 궤적 $p_{old}(u)$ 의 가장 가까운 점을 뜻한다. 예를 들어서 Fig. 7(a)에서 보는 바와 같이 차량 중심으로부터 η 만큼 떨어진 2개의 주행 궤적 후보상의 점과 과거의 주행 궤적상의 점은 각각 $p_1(u_{s^*})$, $p_2(u_{s^*})$, $p_{old}(u_{i^*})$ 이며 2개의 주행 궤적 후보 중에서 식 (10)을

(a) Consideration of steering actuator constraint

(b) Consideration of constraint and lateral deviation

Fig. 7 Trajectory selection

만족되는 경우인 빨간색 점선은 제외되는 주행 궤적 후보이다.

최종적으로 모든 제한조건을 만족하는 주행 궤적 후보 중 좌우로 적게 흔들리는 궤적을 최종 주행 궤적을 선택하기 위해서 최적조건으로 최대 횡방향 변화량을 최소화 하는 주행 궤적을 선택하였다.

$$t^* = \arg \min_{0 \leq t \leq v} \max_s \| p_t(u_s) \perp \overline{p_0 p_6} \| \quad (11)$$

여기서 $\perp$ 는 점과 직선간의 최단거리선분을 뜻하며 $\overline{p_0 p_6}$ 는 식 (6)에서 정의된다. 예를 들어서, Fig. 6(b)에서 생성된 11개의 궤적 후보에 대하여 식 (10)-(11)를 적용한 후 최종 주행 궤적이 선택되었음을 Fig. 7(b)에서 확인할 수 있다. 선택된 주행 궤적이 횡방향의 변화량이 가장 적음을 알 수 있다.

4. 실험적 검증

두 대의 정지차량이 나란히 서 있는 시나리오와 여러 개의 콘과 드럼통이 불규칙하게 배열되어 있는 복합 장애물 시나리오에 대해서 제안된 알고리즘을 실험적으로 검증하고자 한다. 우선 Fig. 8(c)에서 보는 바와 같이 2대의 정지 차량이 있는 시나리오의 경우 Fig. 8(a)에서 보는 바와 같이 x축 방향으로 약 415(m)되는 위치에서 차량(vehicle)에 대해 라이다가 여러 개의 장애물로 인지하고 있음을 확인할 수 있다(Fig. 8(a)의 십자가 표시 참조). 다음으로 정지 경유점까지의 주행 궤적 상(검은색 일점 쇄선 참조)에 정지 차량이 놓여 있는 것으로 판단이 되어 주행모드가 장애물 회피로 선정이 된다.

다음으로 동적 경유점 생성은 식 (4)와 (5)에 의거하여 5개가 생성되고 식 (1)에 의해 최종적으로 두 개가 각각 생성된다(Fig. 8(a)의 사각형 참조). 동적 경유점이 생성된 후에는 식 (6)-(9)에 의해 2 개의 주행 궤적 후보가 생성된다(Fig. 8(a)의 빨간색 점선 참조). 마지막으로 식 (11)에서 제안한 알고리즘에 의거하여 2개의 주행 궤적 후보 중 하나가 선택된다(Fig. 8(a)의 빨간색 실선 참조).

Fig. 8(c)의 두 번째 그림과 같이 첫 번째 정지 차량을 회피한 후에는 Fig. 8(b)에서 보는 바와 같이 새로운 위험 장애물 즉 두 번째 정지 차량(빨간색 십자

가 표시 참조)이 인지된 것을 확인할 수 있다. 앞서 설명한 내용과 비슷하게 두 개의 동적 경유점 및 주행 궤적 후보가 생성되고 최종적으로 주행 궤적이 선택되고 있음을 확인할 수 있다. 마지막으로 Fig. 8(b)와 (c)에서 보는 두 대의 정지 차량을 모두 회피했음을 보여주고 있다(Fig. 8(b)에서 차량의 위치 참조).

두 번째 장애물 회피 시나리오로 Fig. 9(d)에서 보는 바와 같이 다수의 콘과 드럼통이 배열되어 있는 복합 장애물 시나리오에 대한 회피 궤적 생성 및 주행의 결과를 Fig. 9(a)-(c)에서 확인할 수 있다. 먼저 Fig. 9(a)에서 보는 바와 같이 라이다에 의해서 y축으로 약 -10(m), -30(m), -35(m) 지점에서 장애물이 인지되고 이는 다시 식 (3)-(5)에 의하여 1, 2, 3계층으로 구분된 동적 경유점이 총 6개가 생성되는 것을 볼 수 있다(Fig. 9(a)의 사각형 표시 참조). 다음으로 동적 경유점을 지나는 총 7개 주행 궤적 후보군이 생성된다. 식 (10)에서 제안된 제한조건을 만족하는 경우가 3개 주행 궤적 후보에 해당되며(분홍색 점선 참조), 도로 경계 조건을 만족하지 않는 경우가 1개 주행 궤적에 해당된다(분홍색 실선 참조). 식 (10)의 제한 조건을 만족하는 3개의 주행 궤적 후보 중에서 식 (11)의 우선 순위 조건에 의해 최종 주행 궤적이 결정된다(Fig. 9(a)의 빨간색 실선 참조).

Fig. 9(a)의 y축 방향으로 -10(m) 근처의 장애물 즉 1 계층으로 구분된 장애물을 회피한 후 Fig. 9(b)에서 보는 바와 같이 새로운 위험 장애물이 인지되고 다시 장애물 회피를 위한 주행 궤적을 생성한다. 위에서 설명한 바와 유사하게 총 11개의 경유점과 주행 궤적 후보군이 생성된다. 다음으로 식 (10)-(11)에 의해 최종 주행 궤적이 결정된다. 마지막으로 Fig. 9(c)와 (d)에서 보는 바와 같이 주어진 복합 장애물을 모두 회피했음을 보여주고 있다(Fig. 9(c)에서 차량의 위치와 주행 궤적 참조).

5. 결 론

본 논문에서는 하나 또는 다수의 정적 장애물을 회피를 위한 경로 계획 알고리즘을 제안하였다. 경로 계획 알고리즘은 크게 동적 경유점 생성, 주행 궤적 후보군 생성, 그리고 마지막으로 최종 주행 궤적을 선택하는 과정으로 구성되어 있다. 저가 GPS사

Fig. 8 Dynamic waypoint and trajectory generation for avoidance of two stationary vehicles

Fig. 9 Dynamic waypoint and trajectory generation for complex obstacles avoidance

용으로 인한 성능저하를 보상하기 위해서 위치(pose) 예측기를 추가적으로 사용하여 장애물 회피 주행이 가능함을 선보였다. 최종적으로 2013 무인 자율주행 자동차 경진대회의 장애물 회피 미션을 성공적으로 수행함으로써 장애물 회피 판단, 경로 계획, 제어에 이르는 모든 과정이 성공적으로 이루어졌음을 확인하였다.

후 기

본 논문은 산업통상자원부의 기술료지원사업(10043052, 무인 자율주행 자동차 기술개발 촉진)의 지원을 받아 수행되었으며 2013 무인 자율주행 자동차 경진대회에 참여하기까지 실험을 도와준 아주대학교 자동제어실험실의 무인 차량 개발팀(ADAMIII)에게 감사함을 전합니다.

References

- 1) I. Presinger, Daimler Aims to Launch Self-driving Car by 2020, <http://www.reuters.com/article/2013/09/08>, 2013.
- 2) Y. Kuwata, J. Teo, G. Fiore, S. Karaman, E. Frazzoli and J. How, "Real-time Motion Planning with Applications to Autonomous Urban Driving," IEEE Trans. on Control Systems Technology, Vol.17, No.5, pp.1105-1118, 2009.
- 3) S. Thrun, M. Montemerlo, H. Dahlkamp, D. Stavens, A. Aron, J. Diebel, P. Fong, J. Gale, M. Halpenny, G. Hoffmann, K. Lau, C. Oakley, M. Palatucci, V. Pratt, P. Stang, S. Strohband, C. Dupont, L. Jendrossek, C. Koelen, C. Markey, C. Rummel, J. V. Niekerk, E. Jensen, P. Alessandrini, G. Bradski, B. Davies, S. Ettinger, A. Kaehler and A. Nefian, "Stanley: The Robot that Won the DARPA Grand Challenge," J. of Field Robotics, Vol.23, No.9, pp.661-692, 2006.
- 4) Y. Han, Research of Obstacle Decision and Path Planning for Stable Path Generation of Unmanned Ground Vehicle, Ph. D. Dissertation, Kookmin University, Seoul, 2012.
- 5) K. Chu, Real-time Path Planning of Autonomous Vehicles Based on the Curvilinear Coordinate System, Ph. D. Dissertation, Hanyang University, Seoul, 2011.
- 6) S. Waydo and R. M. Murray, "Vehicle Motion Planning Using Stream Functions," Int. Conf. on Robotics and Automation, pp.2484-2491, 2003.
- 7) S. Gidel, P. Checchin, C. Blanc, T. Chateau and L. Trassoudaine, "Pedestrian Detection and Tracking in an Urban Environment Using a Multilayer Laser Scanner," IEEE Trans. on Intelligent Transportation Systems, Vol.11, No.3, pp.579-588, 2010.
- 8) B. Song, "Cooperative Lateral Vehicle Control for Autonomous Valet Parking," Int. J. Automotive Technology, Vol.14, No.4, pp.633-640, 2013.
- 9) A. Moran and M. Hayase, "Shortest-trajectory Control of Autonomous Mobile Robots Using Nonlinear Observers," Proceedings of the 34th SICE Annual Conference, pp.1415-1418, 1995.
- 10) 이웅희, 윤상열, 심인욱, 신승학, 최종원, 오지원, 심현철, 권인소, 최세범, "KAIST Eure-Car: 복잡한 도로 상황에서의 환경인식 및 충돌회피 가능한 무인자율주행차량 개발," 한국로봇학회, Vol.10, No.2, pp.20-31, 2013.