

# 자율주행 장치를 위한 수정된 유전자 알고리즘을 이용한 경로계획과 특징 맵 기반 SLAM

## Path-planning using Modified Genetic Algorithm and SLAM based on Feature Map for Autonomous Vehicle

김정민 · 허정민 · 정승영 · 김성신\*

Jungmin Kim, Jungmin Heo, Sungyoung Jung and Sungshin Kim\*

부산대학교 전자전기공학과

### 요 약

본 논문에서는 자율주행 장치의 효율적인 자율주행을 위한 특징 맵 기반 SLAM(simultaneous localization and mapping)과 수정된 유전자 알고리즘을 이용한 경로계획을 제안하였다. 현재 연구되고 있는 자율주행 장치들에 있어서 가장 큰 문제점 중 하나는 환경 적응성이다. 이는 새로운 환경에서 자신의 위치를 인식해야 하는 경우와 “kid napping” 문제와 연계되어 자율주행 장치가 새로운 위치 혹은 알려지지 않은 위치에서 자신의 위치를 인식해야 하는 경우로 구분된다. 본 논문에서는 이러한 환경 적응성 문제를 해결하기 위해 초음파 센서를 이용한 특징맵 기반 SLAM을 적용하였으며, 지능형 자율주행 장치의 효율적인 주행을 위해 수정된 유전자 알고리즘(genetic algorithm: GA)을 적용한다. 본 논문에서는 성능을 분석하기 위해 직접 설계·제작한 자율주행 장치를 대상으로 임의의 위치에서 자율주행 장치 스스로 자신의 위치를 인식한 후, 주어진 작업을 수행하기 위해 유전자 알고리즘을 통하여 최적화 된 경로를 따라 주행하는 가를 실험하였다. 실험 결과, 빠르고 최적화된 경로계획과 효율적인 SLAM이 가능함을 확인 할 수 있었다.

**키워드** : 자율주행 장치, 위치추정, SLAM, 유전자 알고리즘, 경로계획

### Abstract

This paper is presented simultaneous localization and mapping (SLAM) based on feature map and path-planning using modified genetic algorithm for efficient driving of autonomous vehicle. The biggest problem for autonomous vehicle from now is environment adaptation. There are two cases that its new location is recognized in the new environment and is identified under unknown or new location in the map related kid-napping problem. In this paper, SLAM based on feature map using ultrasonic sensor is proposed to solved the environment adaptation problem in autonomous driving. And a modified genetic algorithm employed to optimize path-planning. We designed and built an autonomous vehicle. The proposed algorithm is applied the autonomous vehicle to show the performance. Experimental result, we verified that fast optimized path-planning and efficient SLAM is possible.

**Key Words** : autonomous vehicle, localization, SLAM, genetic algorithm, path-planning

## 1. 서 론

최근, 지능형 자율주행 장치는 성장 동력 산업으로 선정되어 광범위한 분야에 적용되고 있으며, 지능형 자율주행 장치의 기술 중에서도 무인자율주행 기술에 관한 연구가 활발히 진행되고 있다[1,2]. 이러한 무인자율주행이 가능한 자율주행 장치는 스스로 작업을 수행하기 위하여 공간에 대한 정보의 수집이 가능해야하며, 자율주행 장치의 위치추정 기술과 전역 경로계획, 주행 중 주변 환경 변화에 대한 지역 경로계획이 필요하다.

전역 경로계획이란 알려져 있는 환경 기반 정보에 근거하여 시작점에서 목표점까지의 경유점을 결정하는 경로계획 방법이며, 지역 경로계획은 각 경유점 사이에서 알려지지 않은 장애물을 만나거나 예상치 못한 임무 변경이 발생할 경우 경로의 재계획이 가능한 경로계획 방법이다. 전역 경로계획의 경우, 환경에 대한 정보를 정확히 가지고 있기 때문에 최적화된 경로를 계획 할 수 있지만, 많은 연산과 저장 공간을 필요하므로 실시간 연산이 필요한 자율주행 장치에 적용하기에는 어렵다. 반면, 지역 경로계획은 최적화된 경로계획을 할 수는 없지만, 비교적 적은 연산과 공간을 필요로 하기 때문에 실시간으로 적용이 가능하나 지역 최소화(local minimum)에 빠질 위험이 크다[3,4]. 이러한 문제를 해결하기 위해 많은 연구가 이루어졌고, 기존의 대표적인 경로계획 방법들로는 A\* 알고리즘(A\* algorithm), 기울기 하강 (gradient descent), 유전자 알고리즘(genetic algorithm) 방법이 있다. 이들 경로계획 방법들의 비교는 표 1과 같다.

접수일자 : 2009년 1월 20일

완료일자 : 2009년 5월 22일

\* Corresponding Author : sskim@pusan.ac.kr

이 논문은 2009년 정부(교육과학기술부)로부터 지원받아 수행된 연구임

(지역거점연구단육성사업/차세대물류IT기술연구사업단)

표 1. 경로계획 방법들의 비교

Table 1. Comparison of path planning methods

	A* Algorithm	Gradient descent	Genetic algorithm
merit	· Optimization · High reliability	· Fast operation	· Optimization · Fast operation
defect	· Very slow	· High occurrence of local-minimum	· Variable performance by parameters

표 1에서 A\* 알고리즘의 경우, 전역적인 최적 경로의 탐색이 가능하고 높은 신뢰성을 보장하지만, 연산에 의한 수행 시간이 오래 걸리는 단점을 가진다[5]. gradient descent 방법은 전체 맵 생성 후, 축소된 검색 영역을 탐색하므로 실시간 연산이 가능하지만 지역 최소화 현상이 빈번히 일어나기 때문에 실제 자율주행 장치에 적용하는데 어려움을 가진다[6]. 이에 반해 genetic algorithm은 빠른 연산 수행 시간과 최적화가 가능하며 장애물 탐색 시간이 빠른 장점을 가지지만 초기에 설정하는 시스템의 parameter 값에 의한 성능 변화의 단점이 있다[7,8].

지능형 자율주행 장치의 핵심기술인 자기위치인식 기술은 맵 매핑 기술과 함께 simultaneous localization and mapping (SLAM) 기술로 명명되어 지고 있으며, 이 기술은 맵 생성 기술에 의해 생성된 맵 정보를 이용하여 위치 측정 기술의 성능을 향상시킬 수 있는 기술이다[9]. 이 기술은 크게 맵 생성과 맵 매핑, 위치측정 기술을 필요로 한다. 이 중 맵 생성 기술과 맵 매핑 기술은 레이저 센서, 비전 센서, 적외선 센서, 초음파 센서 등과 같은 거리 센서를 이용해 활발히 이루어지고 있다[10,11]. 이들 센서들은 각각의 장단점을 가지고 있다. 레이저 센서는 높은 정밀도에 비해 고가이며 고전력이 요구되고 비전 센서의 경우는 많은 양의 환경 정보를 수집할 수 있지만, 타 센서들에 비해 처리해야 하는 데이터 량이 많아 연산 비용이 증가되고 정밀도의 변량 폭이 크다. 적외선 센서는 저가이지만, 최대 측정 거리가 100cm 이하이므로 높은 성능을 기대하기 어렵다. 초음파 센서는 저가에 평균 300cm이상의 거리 측정이 가능하며, 외란에 대한 영향이 적어 높은 정확도와 신뢰성을 가지고 있어 가장 많이 사용되고 있다.

현재까지 개발되어진 자율주행 장치에서 가장 큰 문제점 중 하나는 환경 적응성이다. 이는 새로운 환경에서 자신의 위치를 인식해야하는 경우와 “kid napping” 문제와 연계되어 자율주행 장치가 새로운 위치 혹은 알려지지 않은 위치에서 자신의 위치를 인식해야하는 경우로 구분된다. 본 논문에서는 이러한 환경 적응성 문제를 해결하기 위해 초음파 센서를 이용한 특징 맵 기반 SLAM을 적용하였으며, 지능형 자율주행 장치의 작업을 위한 경로를 최적화하기 위한 방법으로 수정된 유전자 알고리즘을 적용한다.

본 논문은 크게 3단계로 이루어진다. 첫 번째 단계는 맵을 생성하고 생성된 맵에서 특징을 추출하는 단계이고, 두 번째 단계는 자율주행 장치가 스스로 자신의 위치를 인식하기 위한 SLAM에 관한 단계이다. 자율주행 장치는 위치를 인식하게 되면 자신의 위치를 서버로 보낸다. 마지막 단계는 서버가 자율주행 장치의 효율적인 주행을 위해 경로계획을 하는 단계이다.

## 2. 자율주행 장치

본 논문에서는 자율주행이 가능한 장치를 제작하기 위하여 초음파 센서 5개와 엔코더 2개, 자이로, 전자 나침반을 사용하였다. 그림 1은 위치측정을 위해 구현된 자율주행 장치의 시스템 구성을 보여준다.

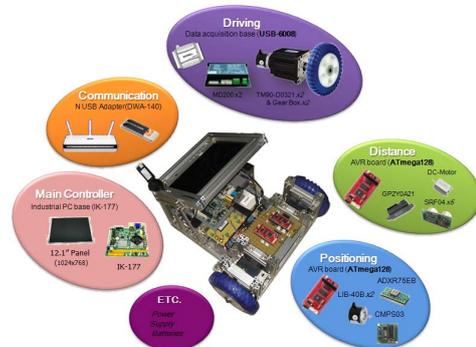


그림 1. 자율주행 장치의 시스템 구성

Fig. 1. System components of autonomous vehicle

구성된 자율주행 장치의 시스템 중 맵 생성과 맵 매핑을 위해 사용된 거리 센서인 초음파 센서와 위치측정을 위해 사용된 엔코더, 자이로, 나침반 센서의 주요 사양은 다음과 같다.

표 2. 사용된 센서들의 주요 사양

Table. 2 Spec. of sensors used

Item	Specification
Ultrasonic (SRF04)	Frequency : 40kHz
	Range : 3~300cm
Encoder (MD200)	Voltage : 5V
	R-F Time : 300us
Gyro (ADXRS401)	Voltage : 5V
	sensitivity : ±75°/s
Compass (CMPS03)	Resolution : 0.1°
	Accuracy : 3~4° approx.

사용된 센서들은 모두 ATmega128을 이용하여 제어하였으며, 센서들 중 초음파는 100ms당 하나의 센서가 계측되도록 설계하였다. 또한 위치측정을 위한 전자 나침반과 엔코더, 자이로는 100ms 속도로 데이터를 통신하도록 설계 하였다. 구동 방식 설계는 제어가 용이하도록 하기위해 그림 2와 같이 선형 2 자유도를 가지는 자동차 구동 방식으로 설계하였다.

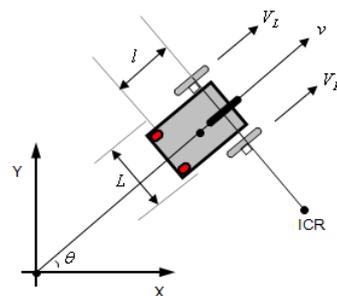


그림 2. 구동 기구학

Fig. 2 Kinematics

### 3. Simultaneous Localization and mapping

#### 3.1 맵 생성

SLAM을 구현하기 위해서는 먼저 맵 생성이 이루어져야 하며, 맵 생성 정밀도에 따라 그 성능이 크게 좌우되게 된다. 본 논문에서는 구조적 맵 생성(hierarchical map building) 기법을 사용하였다[12]. 구조적 맵 생성 기법은 자율주행 장치가 위치한 환경을 나누어 맵 생성 후, 생성된 맵을 연결하는 방법으로 본 논문에서는 다음 그림과 같이 맵을 나누어 맵 생성을 하였다.

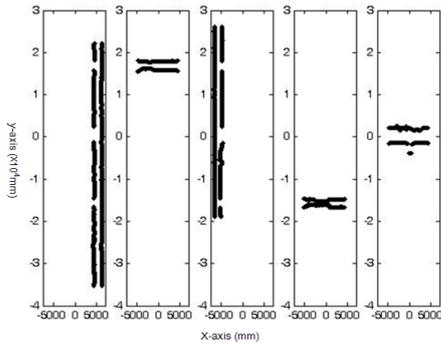
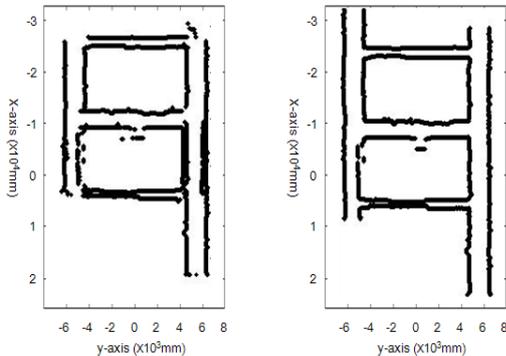


그림 3. 구조적 맵 생성 순서  
Fig. 3 Sequence of hierachial map building

구조적 맵 생성은 그림 3에서 보이는 바와 같이 맵을 나누어 각각의 정보를 받아 연결하면, 그림 4(b)와 맵이 생성되어진다.



a. 일반적인 맵 생성      b. 구조적 맵 생성  
a. General map building    b. Hierarchical map building

그림 4. 맵 생성의 비교  
Fig. 4 Comparison of map building

구조적 맵 생성은 일반적인 맵 생성(그림 4(a))에 비해 중첩된 벽과 센서의 오차가 적어 더욱 정확한 맵을 만들 수 있다. 하지만 이와 같이 그리드 형태로 맵을 표현할 경우에는 매우 큰 저장 공간을 필요로 하여 실시간 연산이 필요한 자율주행 장치에 적용이 어렵다. 이에 본 논문에서는 실시간 연산을 위해 특징 맵으로 변환하여 사용한다. 특징 맵을 구성하는 방법은 연속된 점들의 변화량의 크기에 따른 것으로 변화량이 작은 것(문, 거울 등)과 변화량이 너무 큰(통로)의 특징을 이용한다. 찾아진 특징들은 각 특징들의 종류와 위치한 좌표를 록업 테이블로 저장 후, 사용하였다.

#### 3.2 사용된 SLAM

임의의 위치에서 자율주행 장치가 목표지점으로 가기 위해서는 자율주행 장치가 위치한 맵 상의 절대위치를 알아야 한다. 이는 SLAM을 통해 가능하다. 임의의 위치에서 맵 상에 자율주행 장치의 절대위치를 구하기 위한 SLAM와 전처리 알고리즘은 다음과 같다.

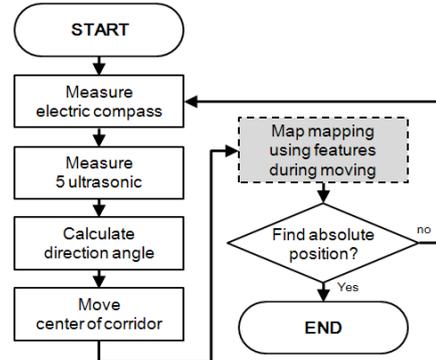


그림 5. SLAM과 전처리 알고리즘  
Fig. 5 SLAM and preprocess algorithm

임의의 위치에서 자율주행 장치는 전자 나침반을 통해 계측된 각도와 초음파 값을 통해 상대적인 방향각을 계산한다. 그리고 계산된 방향각과 생성된 맵 정보를 이용해 복도의 중앙에 이동한다. 자율주행 장치가 복도의 중심으로 이동하는 중에 초음파를 통해 특징을 찾게 되면(점선 박스), 다음 알고리즘과 같이 SLAM을 시작한다.

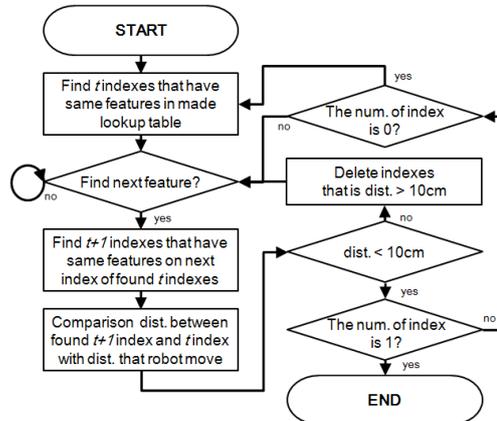


그림 6. SLAM의 맵 매핑 알고리즘  
Fig. 6 Map mapping algorithm of SLAM

위 알고리즘은 자율주행 장치가 초음파를 통해 특징점을 찾게 되면, 3.1절에서 만든 특징 맵의 록업 테이블에서 같은 형태의 특징을 가진 모든 t번째 인덱스를 찾는다. 그리고 자율주행 장치가 다음 특징점을 찾으면, t번째 인덱스의 다음 t+1 번째 인덱스들을 찾게 된다. 찾아진 t+1번째 인덱스는 t번째 인덱스와 t번째 인덱스의 특징 좌표 값을 이용하여 사이의 거리를 계산하고, 이 거리가 10cm 이상인 인덱스들을 삭제 한다. 만약 10cm 이하이고, 찾은 인덱스의 개수가 하나밖에 없다면 이는 절대좌표 값으로 취한다. 만약 인덱스의 개수가 없다면 처음부터 다시 반복하게 되고, 하나 이상 이라면 다음 특징점을 찾을 때까지 기다리는 알고리즘이다.

### 4. 유전자 알고리즘을 이용한 경로계획

#### 4.1 뉴럴-네트워크를 이용한 장애물 체크

시뮬레이션이 아닌 실제 자율주행 장치의 경로계획을 위해서는 빠른 장애물 체크와 용이한 장애물의 삭제·추가 가능해야한다. 이에 본 논문에서는 뉴럴-네트워크 모델(neural network model)을 이용한 장애물 체크를 한다. 뉴럴-네트워크는 활성화 유·무인 2가지 상태로 나타내지므로, 간단한 연산을 통해 장애물 체크가 가능하다[13]. 그림 7과 식 (1)은 뉴럴-네트워크 모델의 이용 예와 인접한 뉴런들의 신호 계산 식을 보여준다.

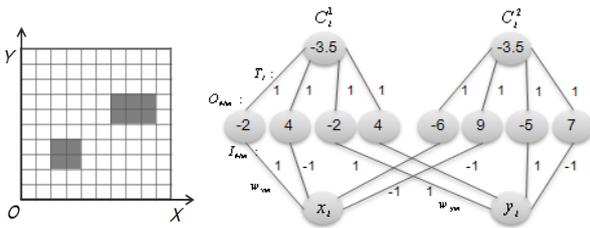


그림 7. 뉴럴-네트워크를 이용한 장애물 체크  
Fig. 7 Obstacle checking using neural network

$$C_i^1 = f(T_i), T_i = \sum_{m=1}^M O_{Mm} + \theta_T,$$

$$O_{Mm} = f(I_{Mm}), I_{Mm} = w_{xm}X_i + w_{ym}Y_i + \theta_{Mm} \quad (1)$$

여기서,  $C_i^1$ 는 네트워크의 출력 레이어의 출력이며,  $T_i$ 과  $\theta_T$ 는 각각 출력 레이어의 입력과 문턱치(threshold)값이다.  $O_{Mm}$ 과  $I_{Mm}$ ,  $\theta_{Mm}$ 은 각각 중간 레이어의 입력 값과 출력 값, m번째 노드의 문턱치 값을 나타낸다. 마지막 수식은  $w_{xm}$ 과  $w_{ym}$ 은 각각 x와 y 입력의 가중치 값이다.

#### 4.2 일반적인 유전자 알고리즘을 이용한 경로계획

유전자 알고리즘은 자연 진화(natural evolution)의 과정을 관찰하여 개발된 최적화 알고리즘이다. 이는 이진 스트링(binary string)으로 코딩된 집단(population)으로 구성된다. 이들을 유전자 연산자인 교배(crossover)와 돌연변이(mutation)를 이용한 재생산(reproduction)을 통해 최적의 해를 찾는다[13]. 일반적인 유전자 알고리즘의 수행 순서는 다음과 같다.

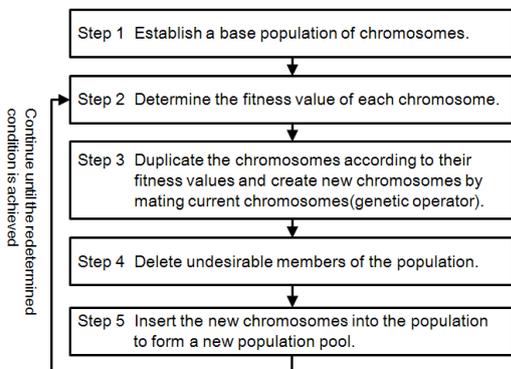


그림 8. 유전자 알고리즘  
Fig. 8 Genetic algorithm

본 논문에서 개체(individual)는 무인 자율주행 장치의 이동 경로를 가지는 좌표의 열을 나타내고, 유전자(gene)는 개체를 구성하는 작업 환경의 좌표 값을 나타낸다. 유전자 풀(genetic pool)은 이러한 개체들을 모아둔 집합체로 부모 개체가 될 수 있는 개체들의 집합 나타낸다.

유전자 알고리즘의 연산 속도향상을 위해 맵의 절대좌표 X-Y 좌표계를 로봇의 방향각을 기준으로 X'-Y' 좌표계로 식 (2)를 통해 변환하여 계산을 쉽게 할 수 있다.

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} \cos\alpha & -\sin\alpha \\ \sin\alpha & \cos\alpha \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} x_s \\ y_s \end{bmatrix} \quad (2)$$

$\alpha$ 는 맵 상의 각도와 로봇의 주행 방향각과의 차이값이며,  $x_s, y_s$ 는 자율주행 장치의 현재 위치좌표이다. 변환된 좌표계는 그림 9와 같다.

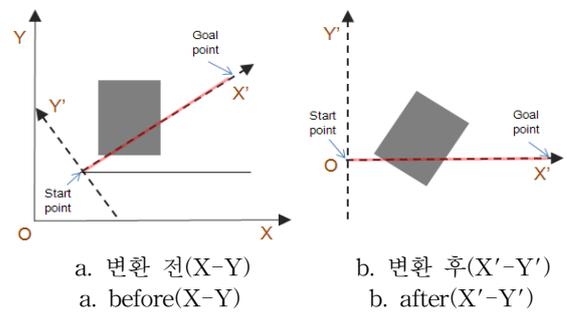


그림 9. 좌표계 변환

Fig. 9 Transformation of coordinate system

일반적으로 유전자 알고리즘을 이용한 경로계획은 시작 위치에서 목표위치를 일정 간격으로 나누고, 이를 후보 경유 지점으로 만들어 수행하게 된다. X'-Y' 좌표계에서 시작위치와 목표위치까지 일정한 간격으로 나눈 지점을 노드점(node point)이라고 하며, 노드점을 기준으로 자율주행 장치가 이동할 수 있는 후보 경유 지점을 노트점(knot point)이라고 한다. 그림 10은 유전자 알고리즘을 이용한 경로계획을 위한 노트점과 노드점을 보여준다.

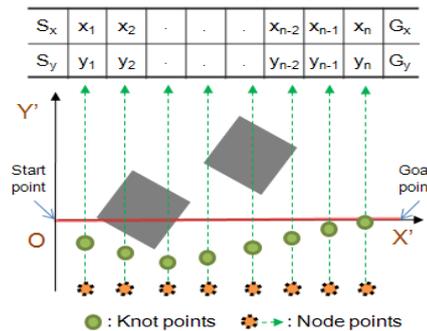


그림 10. 노트점과 노드점

Fig. 10 Knot point and nod point

본 논문에서는 노트점의 간격은 자율주행 장치의 최소 회전 반경을 기준으로 고정값으로 설정하였으며, 노트점의 범위는 작업환경내의 가장 큰 장애물의 장축 길이의 두 배로 설정하였으며, 각 노트점마다의 노트점의 범위는 일정하게 하였다.

각 세대들 간의 우수 형질과 열성 형질을 판별하기 위한 적합도(fitness)를 결정하는 함수는 다음 식과 같이 사용하였다.

$$fitness = \frac{1}{\sum_{i=0}^N \sqrt{(x_{P_i} - x_{P_{i+1}})^2 + (y_{P_i} - y_{P_{i+1}})^2}} \quad (3)$$

위 식에서  $x_{P_i}$ 는 유전자 풀에서  $P$ 번째 개체의  $i$ 번째 노드의  $x$ 좌표 값을 의미하며,  $x_{P_{i+1}}$ 는 다음 노드의  $x$ 좌표 값을 의미한다.  $y_{P_i}$ 는  $P$ 번째 개체의  $j$ 번째 노드의  $y$ 좌표 값을 의미하며,  $y_{P_{i+1}}$ 는 다음 노드의  $y$ 좌표 값을 의미한다.  $N$ 은 개체의 모든 노드 개수를 의미한다.

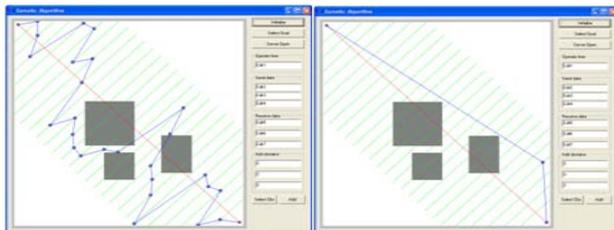
### 4.3 수정된 유전자 알고리즘

4.2절에서 언급한 것처럼, 유전자 알고리즘은 교배, 돌연변이 연산만을 이용하는 데, 이들만을 이용한 경로계획은 최적화를 위해 매우 오랜 시간을 필요로 한다. 이에 본 논문에서는 삭제 연산자를 제안한다. 실제로 유전자 이론에서 삭제 연산자는 돌연변이 연산에 하나로 보고 있다. 이는 자율주행 장치의 동작에 불필요한 부담을 가중시키는 노트를 삭제하는 연산자이다. 삭제 연산자의 수행은 다음과 같다.

```
for i = 1 to (N-1) {
    if ( line_detect(Pi-1 , Pi+1) == 1 ) {
        delete_node(Pi);
        continue;
    }
    else break; // obstacle detected !
}
```

그림 11. 노드 삭제 연산자  
Fig. 11 Node delete operator

여기서  $N$ 은 후보 경유지의 개수를 나타내며, 이전위치  $P_{i-1}$ 점과 다음위치  $P_{i+1}$ 점을 이은 직선상에 장애물이 없을 경우에 가운데 점  $P_i$ 를 삭제하는 방법이다. 이는 자율주행 장치의 동작에 불필요한 부담을 가중시키게 되므로 이러한 노트를 삭제 시켜, 적은 연산만으로도 최적화된 결과를 얻을 수 있다. 그림 12는 50개의 염색체 크기(chromosome size)와 50번의 생산 크기(generation size)의 적은 연산을 수행했을 때, 삭제 연산자를 사용하지 않은 결과와 이용한 결과를 보여준다.



a. 삭제 연산자가 없을 때    b. 삭제 연산자가 있을 때  
a. Without delete operator    b. With delete operator

그림 12. 유전자 알고리즘 결과  
Fig. 12 Result of genetic algorithm

## 5. 실험 및 결과

### 5.1 실험 환경

본 논문에서는 실험을 위해 직접 설계·제작한 실제 자율주행 장치와 관리자, 즉 서버를 이용한다. 제작된 로봇은 산업용 PC 기반으로 제작하였으며, 개발 환경은 Windows XP Fundamental 상에서 Borland C++ 6.0 builder와 Codevision, AVR studio를 사용하였다. 서버 역할로는 Windows XP 상에서 Borland C++ 6.0을 이용하여 개발하였다. 그림 13은 실험에 사용된 자율주행 장치를 보여준다.



그림 13. 실험에 사용된 자율주행 장치  
Fig. 13 Used autonomous vehicle for experiment

실험 방법은 크게 세부분으로 이루어진다. 첫 번째는 추출된 특징을 이용하여 특징 맵을 생성한 후, 그 성능을 실험하는 것이고, 두 번째는 수정된 유전자 알고리즘을 이용한 경로계획의 성능을 평가하는 것이다. 마지막 실험은 임의의 위치에 놓인 자율주행 장치가 생성된 맵 상에서 스스로 자신의 위치를 인식하고, 서버로부터 받은 경로를 따라 주행을 제대로 할 수 있는 지에 대한 실험이다.

### 5.2 특징 맵의 성능 평가

생성된 맵은 연속된 점들의 변화량의 크기에 따른 변화량에 따라 변화량이 작은 특징과 변화량이 큰 특징을 추출을 분류하고, 맵 특징으로 사용한다. 그림 14는 생성된 맵에서 추출된 특징을 보여준다.

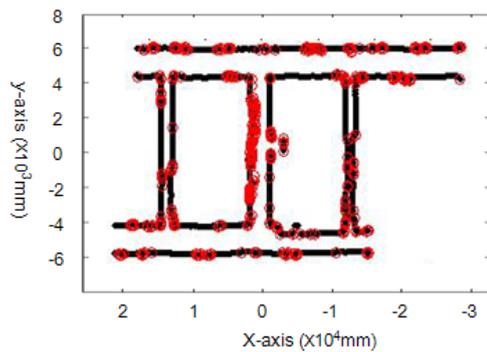
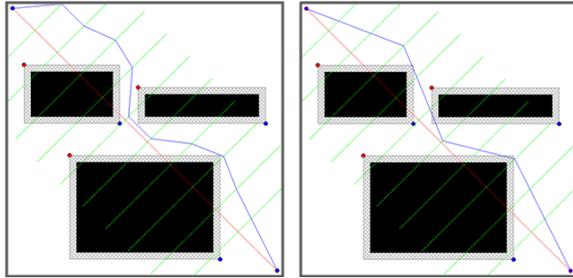


그림 14. 특징 맵  
Fig. 14 Feature map

맵 생성을 통해 찾은 특징은 초음파의 특성의 특성에 의해 중첩된 정보들을 포함한다(그림 14). 이에 본 논문에서는 추출된 특징들은 클러스터링 방법을 이용하여 줄이고, 이를 특징으로 사용하였다. 실험 결과, 찾은 106개의 특징이 추출되었고, 생성된 맵을 기준으로 특징들의 오차는 평균:5.43cm, 분산:21.99cm을 가짐을 확인했다.

5.3 유전자 알고리즘을 이용한 경로계획의 성능평가

본 논문에서는 수정된 유전자 알고리즘의 성능을 평가하기 위하여, 일반적인 유전자 알고리즘과의 성능을 비교하였다. 그림 15는 탐색체와 생산 크기를 각 100으로 설정하여 적은 연산을 통한 각각의 방법에 따른 결과를 보여준다.



a. 일반적인 GA a. GA  
b. 수정된 GA b. Modified GA

그림 15. 경로 계획 결과

Fig. 15 Result of Path-planning

두 방법의 실험 환경은 위 그림과 같이 600cm×600cm 정사각형 맵 상에서 3개의 고정된 장애물을 알고 있다고 가정한다. 고정된 장애물에서 빗금 친 부분은 자율주행 장치의 이동 회전 반경의 크기이며, 그 영역은 이동이 가능한 영역이다. 또한 시작위치와 목표위치는 각각 (10,10)과 (590,590) 고정하였으며, 이는 시작위치와 목표위치까지의 직선거리는 820.243cm이다. 표 3은 각각의 탐색체 크기와 생산 크기를 독립적으로 50번 연산한 경로계획의 최종 경로들의 총 거리값에 평균과 분산을 보여준다.

표 3. 일반적인 GA와 수정된 GA의 비교 결과  
Table. 3 Comparison result of GA and modified GA

Generati on size	Chromos ome size	Mean / Variance	Without delete operation(cm)	With delete op- eration(cm)
50	50	mean	959.222	886.662
		variance	325.441	937.433
50	100	mean	967.987	886.267
		variance	624.370	1259.254
100	100	mean	935.689	869.687
		variance	604.534	849.765
50	100	mean	959.222	883.100
		variance	325.441	372.380

유전자 알고리즘은 확률적인 최적화 기법이므로, 파라메타의 변경에 의해 성능이 달라지더라도 그 정확한 성과 지표 분석하기는 힘들다. 하지만 삭제 연산자를 사용한 결과와 사용하지 않은 결과를 비교하였을 때, 삭제 연산을 사용한 결과가 더 최적화되었음을 확인할 수 있다.

경로계획의 성능을 경로계획의 최단거리로 생각할 수 있지만, 실제 자율주행 장치에 경로계획을 적용하기 위해서는 최적 경로계획 시간이 매우 중요하다. 그 이유는 자율주행 장치가 주행하는 동안 환경은 매시간 바뀌게 되고, 자율주행 장치의 각 상황에 따라 경로의 재계획이 필요하기 때문이다. 그림 16은 100개의 탐색체 크기와 100번의 생산을 통한 경로계획을 수행했을 때, 매 생산 시의 생성되는 경로들의 거리값을 보여준다.

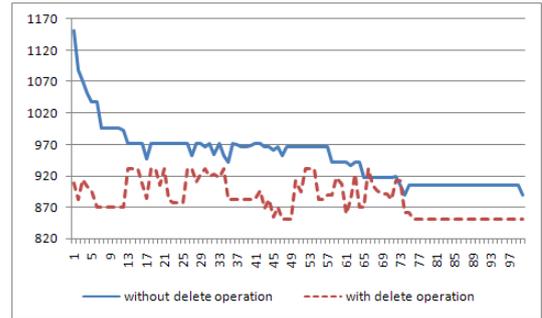


그림 16. 수렴속도 비교

Fig. 16 Comparison of convergence speed

5.4 SLAM의 성능평가

실험 방법은 특징 맵을 이용하여 그림 17의 ㉠-㉦ 구역 8 곳에 임의로 자율주행 장치를 두고, 자율주행 장치가 맵 상의 절대위치를 스스로 찾을 수 있는지 여부를 확인하였다.

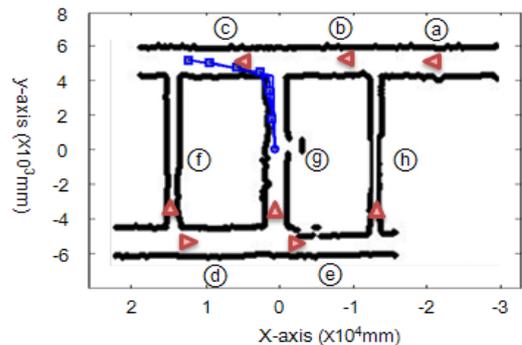


그림 17. 자율주행 장치의 임의의 위치

Fig. 17 Positions of autonomous vehicle on experiment

위 그림에서 ㉢ 구역의 선은 자율주행 장치가 맵 상의 절대좌표를 찾고, 서버에게 경로를 받아 맵의 중심으로 이동하는 것을 보여준다. 자율주행 장치가 노트에 도착할 때마다 자율주행 장치는 자신의 위치를 서버에 전송하고, 서버가 경로를 재계획 하도록 하였다. 8번의 실험에서 모두 절대좌표를 찾음을 확인하였다.

5. 결 론

본 논문에서는 초음파와 전자 나침반, 엔코더, 자이로를 이용하여 자율주행 장치를 직접 설계·제작하였고, 자율주행 장치의 효율적인 자율주행을 위한 특징 맵 기반 SLAM과 수정된 유전자 알고리즘을 이용한 경로계획을 적용하였다. SLAM은 환경 적응성 문제를 해결하기 위하여 초음파 센서를 이용해 특징 맵 생성하고 이를 기반으로 구현하였으며, 지능형 자율주행 장치의 효율적인 주행을 위해 수정된 유전자 알고리즘을 적용하여 경로를 최적화 하였다. 실험은 특징 맵의 성능을 실험과 수정된 유전자 알고리즘과 일반적인 유전자 알고리즘의 경로계획 성능을 비교하였다. 또한 특징 맵 기반 SLAM을 이용해 "kid napping" 문제를 해결할 수 있음을 확인하였다. 모든 실험 결과, 자율주행 장치의 효율적인 주행이 가능함을 확인하였다.

참 고 문 헌

[1] Sebastian Thrun, "Probabilistic Algorithms in Robotics," *AI Magazine*, Vol.21, No.4, pp. 93-109, 2000.

[2] X. Yun and Nilanjan Sarkar, "Dynamic Feedback Control of Vehicle with Two Steerable Wheels," *IEEE Inter. Con. on Robotics and Auto.*, pp. 3105-3110, 1996.

[3] Majdi. M, Deldar. M, Barzamini. R, Jouzdani. J, "AGV Path Planning in Unknown Environment Using Fuzzy Inference System," *Fisrst IEEE International Conference on E-Learning in Industrial Electronics*, pp. 64-67, 2006.

[4] Meng Wang, Liu. J.N.K, "Fuzzy Logic Based Robot Path Planning in Unknown Environment," *Proceedings of International Conference on Machine Learning and Cybernetics*, Vol. 2, pp. 813-818, 2005.

[5] Taixiong Zheng and Xiangyang Zhao, "Research on Optimized Multiple Robots Path Planning and Task Allocation Approach," *Proceedings of the IEEE International Conference on Robotics and Biomimetics*, Kunming, 2006

[6] J. Borestein and Y. Koren, "The Vector Field Histogram-Fast Obstacle Avoidance for Mobile robots," *IEEE Trans on Robotics and Automation*, Vol. 7, No. 3, 1991.

[7] Shuhua Liu, Yantao Tian, Jinfang Liu, "Multi Mobile Robot Path Planning Based on Genetic Algorithm," *Intelligent Control and Automation WCICA*, Vol.5, pp. 4706-4709, 2004.

[8] Afsar. F.A., Arif. M., Hussain. M., "Genetic Algorithm Based Path Planning and Optimization for Autonomous Mobile Robots with Morphological Preprocessing," *Multitopic Conference, INMIC '06. IEEE*, 23-24, pp. 182-187, 2006.

[9] Momotaz Begum, George K. I. Mann, Raymond G. Gosine, "An Evolutionary SLAM Algorithm for Mobile Robots," *Proc. of the 2006 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2006.

[10] M.B. Motamedinejad, R. Barzamini, J. Jouzdani, A. Khosravi, "A New Fuzzy Path Planning For Multiple Robots," *International Conference on Information and Automation*, pp.295-300, 2006.

[11] Surmann. H, Huser. J and Wehking. J, "Path Planning for a Fuzzy Controlled Autonomous Mobile Robot," *Proceedings of the Fifth IEEE International Conference on Fuzzy Systems*, Vol. 3, pp. 1660-1665, 1996.

[12] Ho-Duck Kim, Sang-Wook Seo, In-hun Jang, and Kwee-Bo Sim, "SLAM of Mobile Robot in the indoor Environment with Digital Magnetic Compass and Ultrasonic Sensors," *International Conference on Control, Automation and Systems*, Seoul, pp. 87-90, 2007.

[13] ZHU Yongjie, CHANG Jiang, WANG Shuguo, "A New Path-planning Algorithm for Mobile Robot Based on Neural Network," *Proceedings of IEEE TENCON'02*, pp. 1570-1573, 2002.

저 자 소 개



김정민(Jungmin Kim)

2006년 : 신라대학교 컴퓨터공학과 (공학사)

2008년 : 부산대학교 전기공학과 (공학석사)

2008년~현재 : 동대학원 전자전기공학과 박사과정

관심분야 : 지능형 로봇, 지능 시스템, 신호처리  
E-mail : kjm16@pusan.ac.kr



허정민(Jeongmin Heo)

2007년 : 신라대학교 컴퓨터공학과 (공학사)

2007년~현재 : 부산대학교 전자전기공학과 석사과정

관심분야 : 신호처리, 지능 시스템  
E-mail : hjm1700@pusan.ac.kr



정승영(Sungyoung Jung)

2008년 : 부산대학교 전자전기통신공학부 (공학사)

2008년~현재 : 동대학원 전자전기공학과 석사과정

관심분야 : 지능형 로봇, 지능 시스템  
E-mail : pleiades0223@pusan.ac.kr



김성신(Sungshin Kim)

1986년 : 연세대학교 전기공학과(공학석사)

1996년 : Georgia Inst. of Technology, 전기및컴퓨터공학과 (공학박사)

1998년~현재 : 부산대학교 전자전기통신공학부 부교수

관심분야 : 지능 시스템, 지능형 로봇, 고장 예측 및 진단  
Phone : +82-51-510-2374  
Fax : +82-51-513-0212  
E-mail : sskim@pusan.ac.kr