

# 속도 가변을 통한 영상교시 기반 주행 알고리즘 성능 향상

## Improvement of Visual Path Following through Velocity Variation

최 이 삭, 하 중 은\*  
(I-Sak Choi<sup>1</sup> and Jong-Eun Ha<sup>1</sup>)

<sup>1</sup>Seoul National University of Science and Technology

**Abstract:** This paper deals with the improvement of visual path following through velocity variation according to the coordinate of feature points. Visual path follow first teaches driving path by selecting milestone images then follows the route by comparing the milestone image and current image. We follow the visual path following algorithm of Chen and Birchfield [8]. In [8], they use fixed translational and rotational velocity. We propose an algorithm that uses different translational velocity according to the driving condition. Translational velocity is adjusted according to the variation of the coordinate of feature points on image. Experimental results including diverse indoor cases show the feasibility of the proposed algorithm.

**Keywords:** autonomous navigation, path follow, feature matching, intelligent robot

### I. 서론

로봇의 자율 주행을 위해서는 레이저 거리 센서, 초음파 센서, 카메라, GPS, 관성센서 및 카메라 등의 다양한 센서들이 사용되고 있으며 Urban Challenge 대회를 통해 도심 속의 일상적인 운행환경과 비슷한 환경에서도 자율 주행이 가능함이 입증되었다[1].

자율 주행을 위한 가장 일반적인 방법은 SLAM (Simultaneous Localization And Mapping)으로 불리며 레이저 거리 센서를 이용한 실내 및 실외에서의 다양한 환경하에서의 자율 주행이 가능함이 보여졌다[2,3]. 이들 접근 방법에서 레이저 거리 센서가 주요 센서로 이용되고 있다. 레이저 거리 센서의 경우 다른 센서에 비해 정확한 정보를 제공하는 장점이 있으나 고가격과 처리 시간의 단축의 어려움 등의 단점이 따르게 된다. 이러한 문제를 해결하기 위해 최근에는 상대적으로 저비용이면서 풍부한 정보를 제공하는 카메라를 이용한 자율 주행에 관한 연구가 활발히 진행되고 있다[4-7]. 카메라의 경우 정보 추출을 위한 추가적인 복잡한 처리 알고리즘이 필요하다는 단점이 있다.

본 연구에서는 먼저 주행 경로를 교시후 이를 실제 주행시 영상 정보를 이용하여 동일한 경로를 유지하는 방법론에 대해 다루도록 한다. 이러한 주행 기법은 영상교시 기반 주행 [8-11]으로 불리고 있다. Chen과 Birchfield [8]는 이미지상의 특징점의 추출 및 추적을 이용한 영상교시 기반 주행 방법을 제안하고 있다. [9]에서는 로봇의 Odometry 정보를 추가로 이용하여 [8]의 방법을 확장하였다. [10,11]에서는 마일스톤 이미지 구축시 특징점의 삼차원 정보를 이용하고 있다. 삼차원 정보의 사용을 통해 보다 강인한 경로 추종 결과를 보이고 있다.

본 연구에서는 영상상의 특징점 정보만을 이용한 영상교

시 기반 주행 방법인 [8]의 방법론을 따르도록 한다. [8]에서는 교시 및 주행시에 미리 지정한 일정한 선속도와 각속도를 이용하고 있다. 다양한 환경하의 실험을 통해 기존 방법[8]의 경우, 지정 속도에 따라 주행 성능이 민감하게 의존함을 알 수 있었다. 본 연구에서는 보다 다양한 환경하에서의 안정적인 주행을 위해 이미지상의 특징치의 좌표값을 반영한 속도 가변 알고리즘을 제안하도록 한다.

### II. Chen&Birchfield의 영상교시 기반 자율 주행 방법

본 절에서는 Chen과 Birchfield [8]가 제안한 영상기반의 경로추종 알고리즘에 대해 살펴보도록 한다. 기본적인 과정은 교시 단계와 주행 단계로 구성된다.

교시 단계: 수동으로 로봇을 조종하여 정해진 경로를 주행하면서 로봇에 장착되어 있는 카메라를 이용하여 이미지를 획득한다. 주행 중에 촬영된 이미지는 일정한 주행 시간의 간격으로 분류된다. 분류된 각 단계의 시작 이미지 내에서 특징점을 추출하고 연속된 마지막 이미지까지 이를 추적하는 과정을 거친다. 여기서 각 단계의 시작 이미지는 마일스톤 이미지로 결정되고 각 특징점들의 초기 위치와 최종 위치 정보도 같이 보존하도록 한다.

주행 단계: 그림 1과 같이 교시 단계에서의 초기 출발 위치 근처에 로봇을 위치시킨다. 그리고 순차적으로 불러들인 마일스톤 이미지 내의 특징점들과 현재 이미지 내의 특징점들을 비교한다. 각각의 특징점 간의 상대적인 위치는 로봇의 진행방향을 결정하는 요소가 된다. 다음은 두 특징점들 간의 상관 관계에 따른 로봇의 조향 방향에 대한 판단식이다[1].

$$\begin{cases} f_i^c > 0 \text{ and } f_i^d < 0 \text{ then turn right} \\ f_i^c > 0 \text{ and } f_i^d > f_i^c \text{ then turn right} \\ f_i^c < 0 \text{ and } f_i^d > 0 \text{ then turn left} \\ f_i^c < 0 \text{ and } f_i^d < f_i^c \text{ then turn left} \\ \text{otherwise do not turn} \end{cases} \quad (1)$$

\* 책임저자(Corresponding Author)

논문접수: 2010. 12. 23., 수정: 2011. 1. 14., 채택확정: 2011. 1. 24.

최이삭: 서울과학기술대학교 NID 융합 대학원

(isakchoi@seoultech.ac.kr)

하중은: 서울과학기술대학교 자동차공학과(jeha@seoultech.ac.kr)

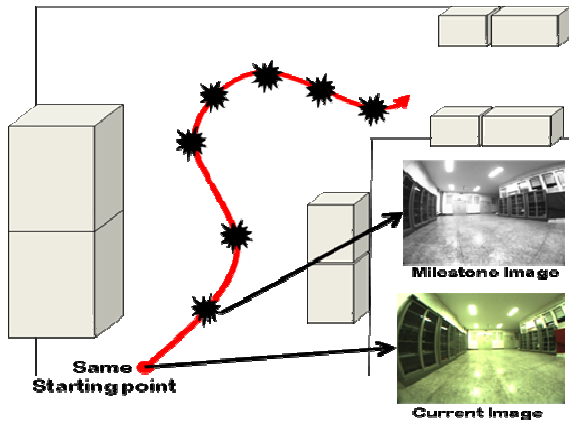


그림 1. 영상교시 기반의 자율 주행.  
Fig. 1. Autonomous navigation using visual path following.

$f_i^c$  는 현재 이미지상의  $i$  번째 특징점의  $x$  좌표값이며,  $f_i^d$  는 마일스톤 이미지상의 대응하는 특징점의  $x$  좌표값이다. 영상상의 특징점 좌표는 이미지 중심을 원점으로 하여 계산한다. 이미지 중심의 오른쪽 부분의  $x$  좌표는 양수이며 왼쪽 부분은 음수이다. [8]에서 제안한 알고리즘은 위와 같이 추출된 특징점들의  $x$  좌표값만을 이용하여 영상 서보잉을 통해 목표지점으로 자율 주행한다. 전진 속도의 경우 일정 값을 유지하며 위와 같이 조향 방향에 대해서만 bang-bang 제어 기법을 통한 목표 지점으로의 도달을 수행하고 있다. 로봇은 순차적으로 해당 단계의 마일스톤 이미지를 통과함으로써 목표 지점에 도달하게 된다.

마일스톤 이미지의 특징점과 현재 이미지의 특징점이 서로 일치할 때 현재 단계는 완료되고 다음 단계의 마일스톤 이미지가 새로운 표적이 된다. 마일스톤 천이 조건은 특징점들 간의  $x$  좌표값의 평균 오차를 이용한다. 현재 특징점이 마일스톤 이미지의 특징점에 접근할수록 오차는 감소하며, 벗어날수록 오차는 증가하게 된다. [8]에서는 오차값이 감소하다가 증가하는 시점을 마일스톤 이미지 변경 시점으로 사용하였다.

III. 제안 방법

기존 알고리즘[8]은 학습시와 실제 주행시 미리 지정한 일정 크기의 선속도와 각속도를 사용하고 있다. 그러나, 보다 다양한 주행 조건하에서의 안정적인 주행을 위해서는 주행 구간 특성에 맞는 주행 속도에 대한 제어가 필요하다. 기존 알고리즘[8]에서 사용하는 로봇과 동일한 모델을 사용하고 있으나 계산 처리 시간, 주행 환경의 차이등으로 인해 동일한 선속도와 각속도 값을 사용하는 것은 어렵다. 따라서, 먼저 다양한 선속도와 각속도의 조합을 통해 기존 알고리즘의 주행 성능을 분석해 보았다.

그림 2는 기존 알고리즘에 의한 다양한 속도 조합에 따른 주행 궤적이다. 주행 궤적은 P3-AT 로봇이 제공하는 자세 정보를 이용하였다. 검은색 선은 교시단계에서의 궤적이고 파랑색 선은 주행시의 궤적이다. 기존 알고리즘의 경우, 표 1과 같이 각속도 7가지, 선속도 5가지 조합으로 이루어진 총 35개의 속도 조합에 대해 15개의 조합만이 주행을 완료하였으

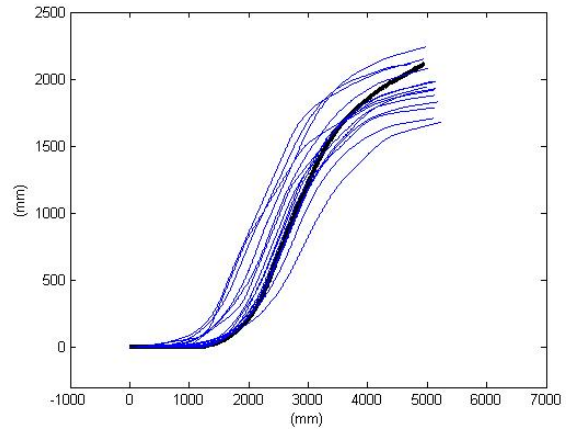


그림 2. 기존 알고리즘[8]의 주행 궤적 비교 (검정색: 교시단계의 궤적, 파랑색: 주행 단계의 궤적).

Fig. 2. Comparison of robot's running locus by previous algorithm[8] (black: locus during teaching, blue: locus during navigation).

표 1. 주행 속도에 따른 주행 오차 (F: 실패 경우).

Table 1 Navigation error according to running velocity (F: failure case).

선속도[mm/s] \ 각속도[deg/s]	30	40	50	60	70
1	F	F	F	F	F
2	F	F	F	F	F
3	516	F	F	F	F
4	229	F	F	F	F
5	188	260	361	433	F
6	258	362	284	126	F
7	174	11	38	251	66

며 나머지 경우에는 실패하였다.

표 1에서 이동로봇의 주행오차는 교시단계의 로봇 종료지점 좌표와 주행단계의 완료지점 좌표간의 거리 차이이다. 주행을 완료한 경우의 주행 오차는 최소11mm, 최대516mm사이의 값을 가진다. 기존 알고리즘[8]의 실내 실험에서 측정된 최대 주행오차 35mm를 견주어 볼 때 주행 오차 11mm인 선속도40mm/s, 각속도 7deg/s의 경우가 적정 속도로 예상된다. 이를 통해 기존 알고리즘의 경우 주어진 주행 조건에 따라 각속도 및 선속도의 조절이 필요함을 알 수 있다.

그림 3은 로봇의 직진 이동 및 제자리 회전 이동에 따른 특징점의 이동 경향이다. 녹색 점은 특징점의 시작 지점, 빨강색 점은 특징점의 종료 지점이다. 카메라가 로봇과 평행하게 장착된 경우, 로봇의 직진 운동시 이미지에 투영되는 특징점은 이미지의 중심으로부터 바깥으로 방사되는 경향을 가진다. 이와 달리 로봇이 제자리 회전을 하는 경우 특징점은 X축 방향으로만 이동하는 경향을 가진다. 이동 로봇의 모든 주행은 직진과 제자리 회전의 조합으로 표현될 수 있으며 특징점의 상대적인 이동 경향을 통해 로봇의 주행 경로를 추측할 수 있다. 그림 3의 두 가지 경우에서 차이점은 회전 구간에 비해 직진 구간에서의 특징점이 Y축 방향으로 이동하는 경향이 크다는 점이다. 이러한 경향을 보이는 이유는 카

메라와 이동로봇의 기하학적 배치에서 알 수 있다. 이동로봇에 장착된 카메라는 그림 4와 같이 카메라의 주축 Z'와 로봇의 전진방향 Z이 서로 평행하도록 설치되어 있다. 로봇이 Z축으로 전진 구동시에 카메라 또한 Z'방향으로 이동한다. 로봇의 전진이동에 따라 특징점은 그림 5와 같이 이미지상에서 Y축으로 이동한다. A는 출발지점, B는 로봇의 도착지점이라고 할 때, 로봇이 A지점에서 B지점으로 전진 이동시 공간상의 점 F는 각각 이미지상의 점 f1, f2로 투영된다. f1, f2는 이미지 상의 특징점 좌표이며 이미지의 중심에서의 거리차이를 d1, d2라고 할 때, 이동 로봇이 공간상의 점 F에 가까워질수록 (d2-d1)은 커짐을 알 수 있다. 그러므로 그림 4와 같이 카메라가 로봇에 평행하게 설치된 경우 특징점의 Y축 방향의 이동 성분과 로봇의 전진 이동은 비례 관계임을 알 수 있다.

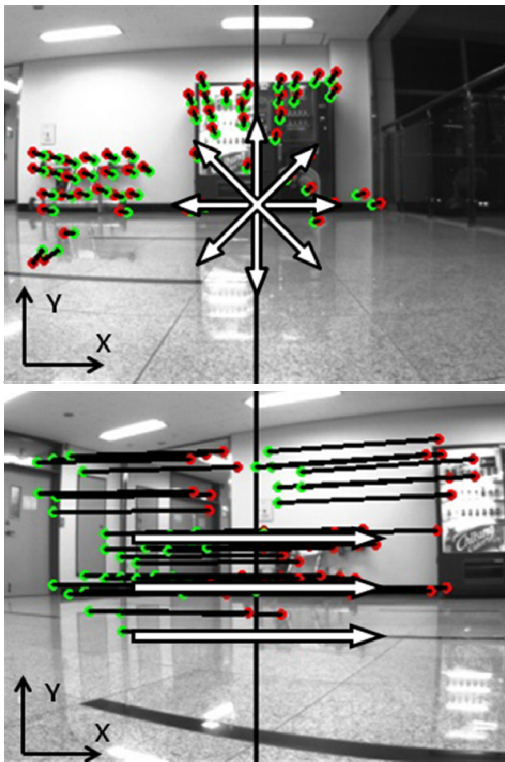


그림 3. 주행 형태에 따른 특징점의 이동 (상: 직진 운동 하: 회전 운동).

Fig. 3. Displacement of feature point according to navigation type (upper: straight motion lower: rotational motion).

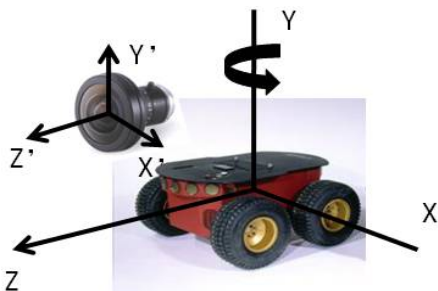


그림 4. 카메라와 로봇의 기하학적 모델.

Fig. 4. Geometric model of a camera and a mobile robot.

그림 3은 등속으로 주행하는 로봇에서 촬영된 이미지상에서 특징점을 추출하여 0.8초 동안의 특징점의 이동 위치를 나타낸 것이다. 추출된 모든 특징점들의 y좌표의 차이(d1-d2)의 평균은 직선 운동하에서 5.2pixel, 회전 운동하에서는 0.9pixel로 나타났다. Y좌표의 차이는 회전 운동시에 비해 직선 운동시에 상대적으로 크다는 것을 알 수 있다. 이와 같은 고찰을 바탕으로, 다양한 환경하에서의 안정적인 로봇의 주행을 위해 특징점의 y좌표 성분을 고려한 가변적인 전진속도 결정 방법을 제안하도록 한다. 다음은 특징점의 y좌표 성분 차이를 고려한 전진속도  $V_i$ 의 결정식이다.

$$V_i = \tanh\left(\frac{d2-d1}{G}\right) * V_M \quad (2)$$

d1은 이전 이미지상의 특징점의 y 좌표값이며 d2는 현재 이미지상의 대응하는 특징점의 y 좌표값이다. G는 y좌표의 평균 차이에 대한 속도의 변화를 결정하는 인자이며,  $V_M$ 은 최대 전진 속도이다.

그림 6은 최대 전진 속도를 60mm/s으로 결정할 때 G값의 변화에 따른 전진속도  $V_i$ 를 보여준다. G값이 클수록 전진속도의 y좌표 차이에 대한 전진속도의 크기 변화가 완만해지는 것을 확인할 수 있다.

제안 알고리즘의 경우 식 (2)를 이용하여 주행시 주행 환경에 따라 전진 속도를 가변 하도록 한다.

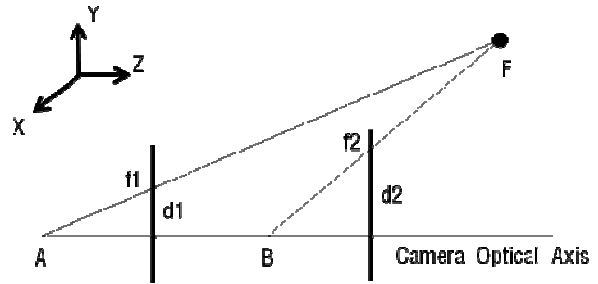


그림 5. 로봇의 전진이동에 따른 특징점의 변화.

Fig. 5. Variation of feature point according to the forward movement of a robot.

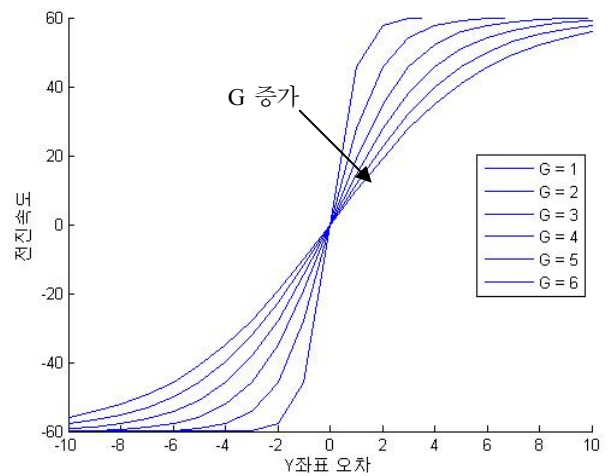


그림 6. y좌표 차이에 대한 속도 그래프.

Fig. 6. A plot of velocity according to the difference of y-coordinate.



그림 7. 실험에 사용한 로봇 (MobileRobot's P3-AT).  
Fig. 7. A robot used in experiments (MobileRobot's P3-AT).

IV. 실험 결과

실험은 MoileRobot사의 P3-AT 이동로봇에 노트북(Dell studio 14)을 장착하여 수행하였다. 카메라는 Point Grey사의 FireFly를 이용하였으며 어안렌즈를 사용하였다. 카메라는 로봇과 평행하게 설치하였다. 획득된 이미지의 크기는 320×240 크기이며, 특징점 추출 및 매칭을 위해서는 KLT 알고리즘[12]을 이용하였다.

실험은 기존 알고리즘 및 제안 알고리즘이 모두 주행에 성공하는 경우와 기존 알고리즘은 실패하고 제안 알고리즘은 성공하는 두 가지 경우에 대해 다루도록 한다. 주행 성능의 분석은 최종 종료 지점의 좌표값의 오차를 이용하도록 한다. 주행오차의 측정은 P3-AT로봇에서 제공하는 자세 정보와 종료지점의 수작업에 의한 실측 정보를 이용하였다. P3-AT의 자세 정보는 엔코더 및 자이로 센서를 이용하고 있으며 바퀴의 슬립이 발생하는 회전 이동에서는 부정확한 위치 정보를 제공한다. 따라서 실측 측정을 통해 이를 보완하도록 한다.

실험 1은 기존 알고리즘과 제안 알고리즘 모두 주행에 성공한 경우이며 그림 2에서의 경우와 동일한 교시 정보를 이용하였다. 기존 알고리즘의 경우 주행시 선속도 40mm/s, 각속도 6deg/s를 이용하였다. 기존 알고리즘 및 제안 알고리즘 모두 동일 경로에 대해 5번의 실험을 수행하였다.

그림 8은 P3-AT이동 로봇의 자세 정보를 이용한 주행궤적이다. 검정색 선은 교시 단계에서의 궤적, 빨강색 선은 기존 알고리즘에 의한 주행 궤적, 파랑색 선은 제안 알고리즘에 의한 주행 궤적이다. 표 2는 각각의 주행 결과에 따른 주행 종료 지점의 오차이다. P3-AT 이동 로봇이 제공하는 주행 정보를 이용한 주행 오차의 평균은 기존 알고리즘의 경우 311mm, 제안된 알고리즘 경우 385mm이다. 그러나, 실 측정을 이용한 주행 오차의 평균은 기존 알고리즘의 경우 79mm, 제안된 알고리즘 경우 49mm로써 제안 알고리즘이 향상된 결과를 제공한다. 종료 지점의 오차 및 주행 시간 등의 비교에서 기존 알고리즘과 제안 알고리즘이 실험 1의 주행조건에서는 비슷한 성능을 보여줄을 알 수 있다.

실험 2는 기존 알고리즘의 경우 주행에 실패하고 제안 알고리즘의 경우 주행에 성공한 경우이다. 특징점 추적에서 최대 특징점의 개수는 150개로 하였으며 마일스톤 이미지 지정을 위한 추출 간격은 0.6s로 제한하였다.

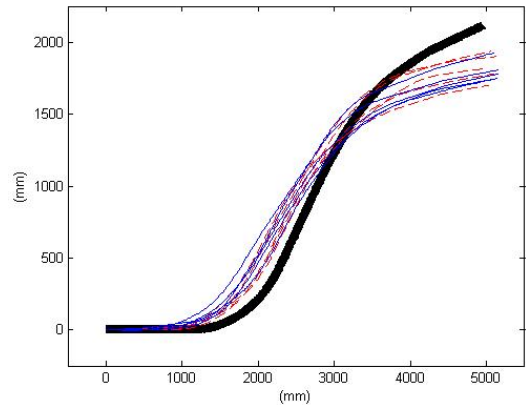


그림 8. 실험 1의 주행 궤적 비교 (검정색: 교시단계, 빨강색: [8], 파랑색: 제안 알고리즘).

Fig. 8. Comparison of the navigation locus of experiment 1 (black: teaching, red: [8], blue: proposed).

표 2. 실험 1의 주행 오차 및 주행시간.

Table 2. Navigation accuracy and time of experiment 1.

		주행 종료지점 (시작위치 원점 기준)		주행 시간 [s]
		P3-AT 제공 좌표 (x,y) [mm]/오차(mm)	실 측정 좌표 (x,y) [mm]/오차[mm]	
	교시	(4953, 2114)	(4743, 1446)	
기존 알고 리즘 [8]	1	(5052, 1936)/203	(4768,1416)/39	187
	2	(5027, 1701)/419	(4707,1350)/102	173
	3	(4997, 1817)/300	(4670,1468)/76	186
	4	(5092, 1779)/362	(4780,1343)/108	195
	5	(5127, 1901)/274	(4790,1495)/67	190
제안 알고 리즘	1	(5150, 1780)/387	(4742,1467)/21	185
	2	(5091, 1926)/232	(4772,1456)/30	182
	3	(5085, 1741)/395	(4727,1342)/103	182
	4	(5154, 1807)/366	(4747,1507)/61	185
	5	(5138, 1748)/409	(4717,1457)/28	184

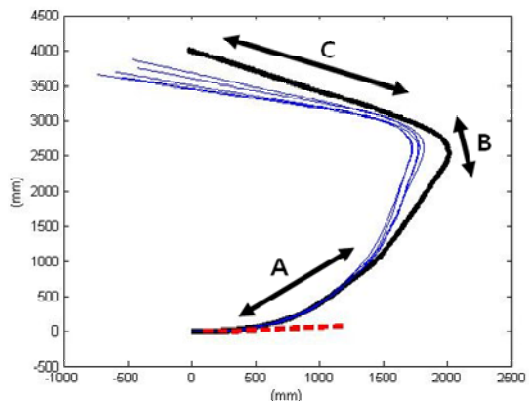


그림 9. 실험 2의 주행 궤적 비교 (검정색: 교시단계, 빨강색: [8], 파랑색: 제안 알고리즘).

Fig. 9. Comparison of the navigation locus of experiment 2 (black: teaching, red: [8], blue: proposed).

그림 9는 실험 2에서의 기존 알고리즘과 제안 알고리즘의 주행 궤적이다. 검은색 선은 교시단계에서 얻은 주행 궤적이며 전체 주행 구간은 A, B, C의 구간으로 구성되어있다. 교시 단계에서 각 구간은 다음과 같은 속도로 주행하였다. A구간

은 회전 반경이 큰 곡선 구간이며 선속도 49mm/s, 각속도 1deg/s로 주행하였다. B구간은 제자리 회전에 가까운 급한 곡선 구간이며 선속도 29mm/s, 각속도 4deg/s로 주행하였다. 마지막 C구간은 직선 구간이며 선속도100mm/s, 각속도 0deg/s로 주행하였다.

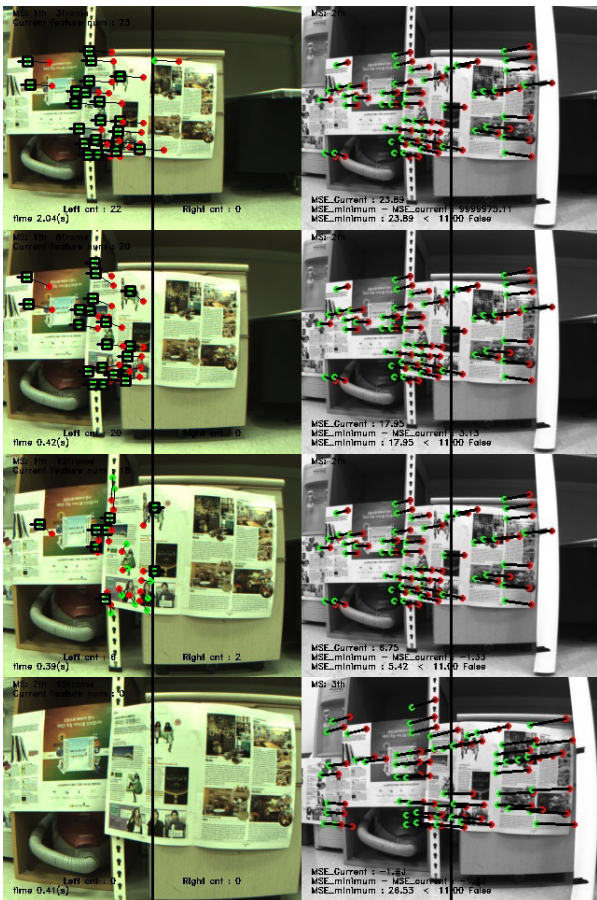
그림 9에서 파랑색 선은 제안된 알고리즘을 이용한 주행 궤적이며 실험은 동일 구간에 대해 총 5회 수행하였다. 빨강색 점선은 기존 논문[8]에 의한 주행 궤적이다. 기존 알고리즘의 경우 주행시 선속도 60mm/s, 각속도 4deg/s로 설정하였다. 실험결과 기존 알고리즘은 A구간에서 경로를 이탈하여 주행에 실패하였고 본 알고리즘은 5번 모두 주행에 성공하였다.

그림 10은 기존 알고리즘이 A구간에서 경로를 이탈하는 과정을 보여주고 있다. (a)는 현재 이미지상의 처리 결과이며 (b)는 마일스톤 이미지이다. 이미지상에서 녹색 점은 현재 특징점이며 빨강색 점은 마일스톤상의 특징점의 위치이다. 검은색 점은 식 (1)의 판단 조건에 의해 왼쪽이나 오른쪽으로 지정된 점이다.

그림 10에서 첫번째 이미지를 보면 현재 특징점과 목표 특징점이 거의 평행하게 위치되어 있다. 이 상황에서 로봇은 왼쪽으로 제자리 회전을 하는 것이 이상적이다. 하지만 고정된 전진속도의 영향으로 로봇은 일정한 회전 반경으로 왼쪽

으로 이동한다. 이 경우의 특징점은 그림 3의 전진이동과 회전 구간이 결합된 양상을 가지게 되며 목표 특징점과 x좌표 방향으로 가까워지지만 y좌표 방향으로는 멀어진다. 세번째 그림은 마일스톤이 변경되기 직전의 이미지이다. 현재 이미지상의 특징점과 목표 특징점이 서로 일치하지 못하였고 이동 로봇이 주행경로를 이탈하게 되는 시점이다. 네번째 이미지는 마일스톤이 변경된 후의 이미지로써 주행경로 이탈로 인해 특징점이 검출되지 않게 된다. 이를 통해 고정된 선속도 및 각속도를 이용한 기존 알고리즘의 경우 다양한 주행 조건에서 보다 안정적인 주행을 위해서는 개선이 필요함을 알 수 있다.

그림 11은 교시 과정에서 구축된 마일스톤 이미지들 중 일부의 이미지들이다. 녹색 점이 현재 마일스톤상의 특징점들의 시작 위치이며 빨강색 점은 종료 위치이다. 그림 12는 제안 알고리즘에 의한 주행중의 결과 이미지이며 녹색 점은 현재 이미지상의 특징점의 위치를 나타내며 빨강색 점은 마일스톤 상의 목표 특징점의 위치를 나타낸다. 검은색 점은



(a) Current image. (b) Milestone image.

그림 10. 기존 알고리즘[8]의 실패 경우.  
Fig. 10. Failure case of previous algorithm [8].

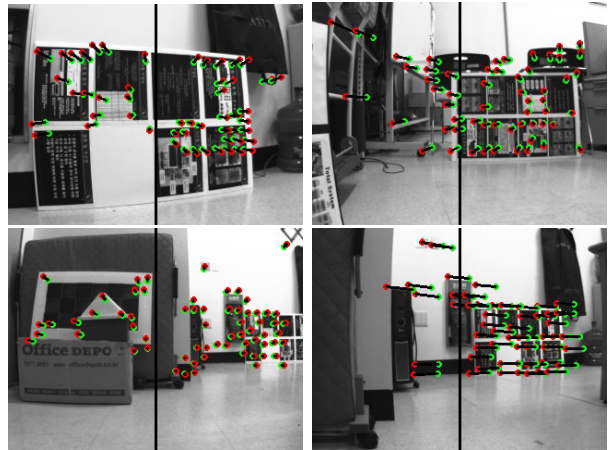


그림 11. 실험 2에서 구축된 마일스톤 이미지들.  
Fig. 11. Some milestone images from experiment 2.

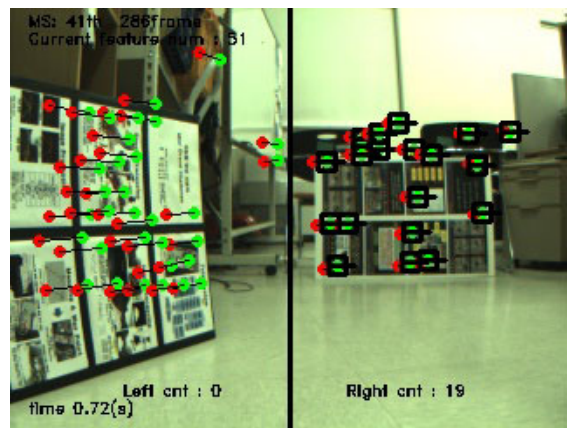


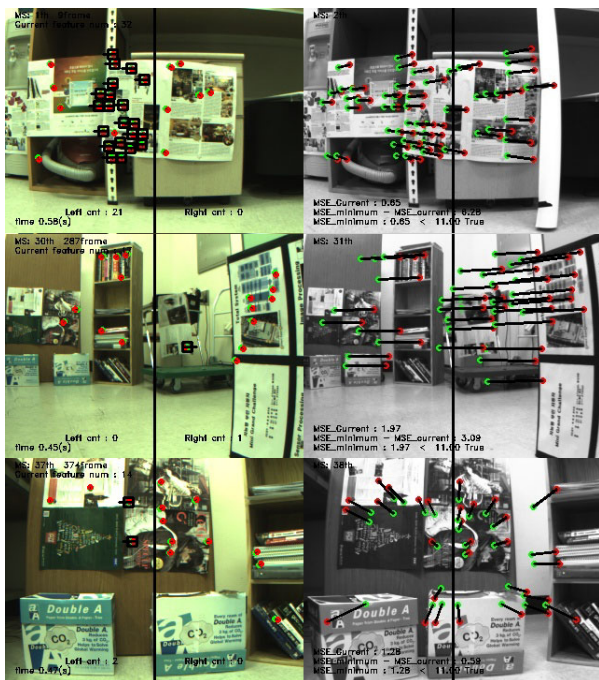
그림 12. 주행중 결과 이미지 (녹색: 현재 이미지상의 특징점 위치, 빨강색: 마일스톤 이미지상의 특징점 위치, 검정: 보팅 결과).

Fig. 12. Result image during running (green: feature location on current image, red: feature location on milestone image, black: voting result).

식 (1)의 판단 조건에 의해 왼쪽이나 오른쪽으로 지정된 점이다. 최종적으로 오른쪽으로 보팅한 점의 개수가 왼쪽 보다 많기 때문에 이와 같은 경우 로봇은 오른쪽으로 주행하게 된다.

그림 13은 제안된 알고리즘이 A, B, C구간을 주행하는 실시간 이미지를 보여준다. 첫번째 이미지는 A구간으로 식 (2)에 의해 최대 선속도 60mm/s 이하로 주행되었다. 두번째 이미지는 B구간으로 실시간 특징점과 목표 특징점간의 Y축의 차이가 거의 없기 때문에 전진이동은 하지 않고 제자리 회전만으로 주행되었다. 마지막 세번째 이미지는 C구간으로 특징점 간의 Y축 차이가 크기 때문에 최대 선속도(60mm/s)로 직진 주행되었다.

표 3은 제안된 알고리즘에 의한 주행 종료지점의 오차이다. 제안된 방법의 주행 종료지점의 오차는 P3-AT 로봇이 제공하는 정보를 이용하는 경우 평균 1297mm이며, 실제 측정



(a) Current image. (b) Milestone image.

그림 13. 제안된 알고리즘에 의한 각 구간에서의 결과 이미지(상:A구간, 중:B구간, 하:C구간).

Fig. 13. Result image from each course by proposed algorithm (top: A course, middle: B course, bottom: C course).

표 3. 실험 2의 주행 오차 및 주행시간.

Table 3. Navigation accuracy and time of experiment 2.

	주행 종료지점 (시작위치 원점 기준)		주행 시간 [s]	
	P3-AT 제공 좌표 (x,y) [mm]/오차[mm]	실 측정 좌표 (x,y) [mm]/오차[mm]		
교시	(4010, -28)	(2906, -838)	89	
제안 알고 리즘	1	(3692, -595)/650	(2911, -871)/32	245
	2	(3860, -467)/463	(2911, -911)/74	258
	3	(3746, -419)/471	(2911, -816)/25	264
	4	(3656, -731)/787	(2925, -1008)/177	245
	5	(3619, -545)/648	(2910, -911)/73	252

에 의한 오차는 75mm이다. 이를 통해 제안 알고리즘의 경우 실험 2의 주행 구간을 안정적으로 주행함을 알 수 있다.

V. 결론

본 논문에서는 이미지상의 특징점들의 좌표만을 이용한 영상교시 기반 알고리즘의 주행 성능 개선을 위한 알고리즘을 제안하였다. 일정한 선속도와 각속도를 사용하는 기존 알고리즘의 다양한 환경하에서의 실험을 통해 영상상의 특징치의 좌표를 이용한 주어진 환경에 따른 선속도 조절 알고리즘을 제안하였다. 실내 환경에서의 실험을 통해 제안한 알고리즘의 주행 성능이 향상됨을 보였다.

참고문헌

- [1] C. Urmson et al., "Autonomous driving in urban environments: boss and the urban challenge," *Journal of Field Robotics*, vol. 25, no. 8, pp. 425-466, 2008.
- [2] M. Montemerlo, S. Thrun, D. Koller, and B. Wegbreit, "FastSLAM: A factored solution to the simultaneous localization and mapping problem," *Proc. of the AAAI National Conference on Artificial Intelligence*, 2002.
- [3] S. Thrun, W. Burgard, and D. Fox, *Probabilistic Robotics*, MIT Press, 2006.
- [4] R. Sim, P. Elinas, M. Griffin, A. Shyr, and J. J. Little, "Design and analysis of a framework for real-time vision-based SLAM using Rao-Blackwellised particle filters," *Proc. of the 3rd Canadian Conference on Computer and Robot Vision (CRV'06)*, 2006.
- [5] S. Se, D. Lowe, and J. Little, "Mobile Robot Localization and Mapping with Uncertainty using Scale-Invariant Visual Landmarks," *The International Journal of Robotics Research*, vol. 21, no. 8, pp. 735-758, 2002.
- [6] A. J. Davison, I. D. Reid, N. D. Molton, and O. Stasse, "MonoSLAM: Real-time single camera SLAM," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, no. 6, pp. 1052-1067, 2007.
- [7] N. Karlsson, E. D. Bernardo, J. Ostrowski, L. Goncalves, P. Pirjanian, and M. E. Munich, "The vSLAM algorithm for robust localization and mapping," *IEEE International Conference on Robotics and Automation*, pp. 24-29, 2005.
- [8] Z. Chen and S. T. Birchfield, "Qualitative vision based path following," *IEEE International Conference on Robotics and Automation*, pp. 2686-2692, 2006.
- [9] Z. Chen and S. T. Birchfield, "Qualitative vision-based path following," *IEEE Transactions on Robotics*, vol. 25, no. 3, pp. 749-754, 2009.
- [10] A. Diosi, A. Remazeilles, S. Segvic, and F. Chaumette, "Outdoor visual path following experiments," *IEEE/RSJ International Conference on Robots and Systems*, pp. 4265-4270, 2007.
- [11] E. Royer, M. Lhuillier, M. Dhome, and J.-M. Lavest, "Monocular vision for mobile robot localization and autonomous navigation," *International Journal of Computer Vision*, vol. 74, no. 3, pp. 237-260, 2007.
- [12] S. Birchfield, "KLT: An implementation of the Kanade Lucas-Tomasi feature tracker," [Online]. Available: <http://www.ces.clemson.edu/~stb/klf/>



### 최 이 삭

2010년 서울과학기술대학교 자동차공학과 졸업. 2010년~현재 서울과학기술대학교 NID 융합대학원 재학중. 관심분야는 지능형 자동차/로봇, 무인 자율주행.



### 하 종 은

1992년 서울대학교 기계공학과 졸업. 1994년 서울대학교 기계공학과 석사. 2000년 한국과학기술원 기계공학과 박사. 2000년~2002년 삼성코닝. 2002년~2005년 동명대학교 멀티미디어공학과. 2005년~현재 서울과학기술대학교 자동차공학과. 관심분야는 지능형 자동차/로봇, 무인 자율 주행, 삼차원 복원, 머신 비전.