어안 워핑 이미지 기반의 Ego motion 을 이용한 위치 인식 알고리즘

Localization using Ego Motion based on Fisheye Warping Image

최 윤 원, 최 경 식, 최 정 원, 이 석 규*

(Yun Won Choi¹, Kyung Sik Choi², Jeong Won Choi³, and Suk Gyu Lee^{4,*})

¹Automotive IT Platform Research Team, ETRI

²Secondary Vocational Education Policy Division, Ministry of Education

³Department of Automatic Electrical Engineering, Yeungnam College of Science & Technology

⁴Department of Electrical Engineering, Yeungnam University

Abstract: This paper proposes a novel localization algorithm based on ego-motion which used Lucas-Kanade Optical Flow and warping image obtained through fish-eye lenses mounted on the robots. The omnidirectional image sensor is a desirable sensor for real-time view-based recognition of a robot because the all information around the robot can be obtained simultaneously. The preprocessing (distortion correction, image merge, etc.) of the omnidirectional image which obtained by camera using reflect in mirror or by connection of multiple camera images is essential because it is difficult to obtain information from the original image. The core of the proposed algorithm may be summarized as follows: First, we capture instantaneous 360° panoramic images around a robot through fish-eye lenses which are mounted in the bottom direction. Second, we extract motion vectors using Lucas-Kanade Optical Flow in preprocessed image. Third, we estimate the robot position and angle using ego-motion method which used direction of vector and vanishing point obtained by RANSAC. We confirmed the reliability of localization algorithm using ego-motion based on fisheye warping image through comparison between results (position and angle) of the experiment obtained using the proposed algorithm and results of the experiment measured from Global Vision Localization System.

Keywords: omnidirectional vision, vision slam, fish-eye lens, ego-motion, optical flow

I. 서론

지능형 로봇에게 SLAM (Simultaneous Localization and Mapping)은 핵심 요소 기술로써 로봇에 부착된 센서를 이용하여 자기 위치를 인식하는 Localization과 주위 환경에 대한 지도를 생성하는 Mapping을 동시에 진행하는 기술이다. 로봇에 대한 연구가 시작된 이후 SLAM은 지도를 이용한 경로 계획, 미지 영역에 대한 탐험 등 로봇의 다양한 활용을 위한 우선 기술이었다. 그 동안 다수의 연구를 통하여 EKF-SLAM, Fast SLAM 등의 이론적으로 완성도가 높은 알고리즘이 나왔지만 다양한 센서 특성과 로봇의 성능에 따라 로봇에 적용이 어려운 것이 현실이다.

SLAM을 위한 센서는 초음파, 레이저 스캐너, RFID, Encoder, 관성센서, 지자기 센서, 이미지 센서 등 다양하게 있다. 사람이 주변 환경 정보를 얻기 위해 오감 중에서 시각을 70-80%를 사용하고 있듯이 로봇도 이미지 센서를 제일 보편적으로 사용하여 왔다. 즉, 로봇은 이미지 정보를 분석하여 주변 특징점을 찾고 특징점 사이의 관계를 기반으로 로봇의

위치를 추정한다. 기존 VSLAM (Visual Simultaneous Localization And Mapping)은 단일 카메라를 이용하는 방법, 스테레오 카메라를 이용하는 방법, 전방향 카메라를 이용하는 방법 등 다양한 방식으로 연구되어 왔다. 단일 카메라 방식은 단일 방향을 60°의 화각으로 정보를 얻기 때문에 지도 생성이 오래 걸리지만 처리속도가 빠른 장점을 가지고 있다. 스테레오 카메라 방식은 동일 특징점을 카메라를 2대에서 측정하여 3차원 깊이 정보를 구하고 이를 바탕으로 위치를 추정하기 때문에 높은 정확성을 가지지만 데이터 처리량이 많은 단점을 가지고 있다. 이에 전방향 카메라 방식은 단일 카메라로 360°의 주변 정보를 한번에 얻기 때문에 처리 속도가 빠르고 많은 정보를 얻을 수가 있다. 또한 로봇이 전진이나 후진에서 자기 위치를 인식하고 경로를 생성하기 용의하다.

기존 전방향 이미지 SLAM은 포물면 반사경(parabol-oidal mirror)에 반사된 이미지를 얻거나[1-3] 천장을 향해 설치한 어안 카메라로 전방향 이미지를 얻었다[4,5]. 그러나 이와 같은 방법은 카메라에 의해 가려지는 부분이 생기거나 조명에 영향을 많이 받는 단점이 있었다. 본 논문에서는 로봇의 상단 부분에 부착된 유리관 말단에 카메라를 바닥 방향으로 설치하여 조명의 영향을 비교적 적게 받으면서 주변 정보를 획득하도록 어안 렌즈를 사용하였다.

기존 Ego-motion은 흔들린 이미지 보상, 카메라의 3차원 좌표에 관한 연구가 진행되어 왔으며, 최근에는 무인자동차의 움직인 위치를 추정하는 분야에 많이 사용되고 있다. 이 알

* Corresponding Author

Manuscript received June 21, 2013 / revised July 31, 2013 / accepted October 4, 2013

최윤원: 한국전자통신연구원 자동차IT플랫폼연구팀

(yunwon.choi@etri.re.kr)

최경식: 교육부 직업교육정책과(oneshei@hanmail.net)

최정원: 영남이공대학교 전기자동차학과(choijw@ync.ac.kr)

이석규: 영남대학교 전기공학과(sglee@yu.ac.kr)

※ 이 논문은 2012년도 정부 (교육과학기술부)의 재원으로 한국연구재단의 기초연구사업 지원을 받아 수행된 것임(2012R1A1B3002240).

고리즘은 연속되는 이미지 정보에서 찾아낸 특징점의 변화에 이용하여 카메라의 모션을 추정하게 된다. 본 논문에서는 전방향 이미지에 동일 선상에 있는 가로축 성분에 대하여 지속적인 방향 벡터를 구하여 로봇이 움직이는 위치를 추정하였다.

본 논문에서 전방향 이미지 SLAM의 성능을 향상시키기 위하여 먼저 바닥을 향해 있는 어안 카메라를 통하여 얻은 전방향 이미지를 워핑 처리를 통하여 정보를 획득하기 용이한 이미지로 변형하고 이미지의 Edge와 가로축에 대한 특징점 정보에 대하여 LKOF (Lucas-Kanade Optical Flow)를 적용하였다. 이렇게 얻어진 방향 벡터 정보에 대하여 최적 수학적 모델을 예측하는 RANSAC을 적용하여 소실점을 추출한다. 이 소실점은 Ego-motion의 FOE (Focus of Expansion)로 이용하여 자기 위치 인식에 사용된다. 이렇게 얻은 FOE에 대한 정보와 방향 벡터의 크기를 기반으로 로봇의 각도와 이동 거리를 추정할 수 있다. 본 논문에서는 제안된 알고리즘을 실제 로봇에 적용하고 특정 경로를 움직이는 동안 위치를 추정한 정보와 전역 위치 인식 프로그램에서 추출한 로봇 좌표를 비교하여 알고리즘의 신뢰성을 검증하였다.

II. 제안한 시스템

1. 전체 구조

본 논문에서 제안한 로봇 시스템은 어안 렌즈를 바닥 방향으로 부착한 카메라를 이용하여 주변 정보를 한번에 획득하고 정보를 처리하여 자기 위치를 판단한다. 그림 1의 순서도는 제안한 알고리즘의 전반적인 구조를 나타낸다. 첫 번째로 로봇에 부착된 어안 카메라를 통하여 얻은 어안 이미지를 역워핑 처리하게 된다. 전처리 된 이미지에서 에지를 검출하고 Ego motion을 위한 가로축 특징점을 추가한다. 특징점에 대해 LKOF를 적용하여 움직임 벡터를 추출한다. 추출한 움직임 벡터의 정보를 비교하여 로봇의 각도를 추출하기 위한 정보와 로봇의 거리를 추출하기 위한 정보로 분리한다. 로봇의 각도를 추출하기 위해 Ego-motion 기법을 적용하여 구한 로봇의 움직인 정도를 바탕으로 로봇의 각도를 추출한다. 로

그림 1. 전체 시스템 블록도.

Fig. 1. The system block diagram.

봇의 거리를 추출하기 위해 움직임 벡터에 대하여 RANSAC 알고리즘을 적용하여 소실점을 구하게 된다. 소실점을 향하는 벡터들의 크기를 기반으로 로봇의 움직인 거리를 추출한다. 이렇게 구한 각도와 거리 정보를 바탕으로 로봇의 위치를 추정한다.

2. 로봇 시스템

본 논문에서 사용한 로봇 시스템은 그림 2에서의 모델링과 같이 2개의 바퀴를 가지고 있으며 각각의 엔코더를 통하여 속도를 얻을 수 있다. 이 모델링을 기반으로 그림 3(a)와 같이 지름이 15cm이고 높이가 30cm인 원통형 소형 로봇을 이용하였다. 이 로봇 가운데에 유리판을 세우고 어안렌즈가 부착된 카메라를 설치하여 주변 이미지를 얻게 된다. 어안 이미지는 그림 3(b)와 같으며 1280x960 크기의 이미지이며 어안렌즈 화각이 185°이기 때문에 주변에 모든 이미지 정보가 한번에 저장된다.

3. 이미지 워핑을 기반으로 한 이미지 전처리

로봇에 부착된 카메라로 들어오는 어안 이미지는 외곽일수록 물체당 픽셀 크기가 작아서 주변 정보를 적게 가지고 있다. 우주 망원경에서 나타나는 이미지 왜곡을 해결하기 위해 개발된 이미지 워핑은 원본 이미지의 픽셀 위치를 바꾸어서 이미지의 형태를 바꿈으로써 획득할 수 있는 정보의 양을 늘리는 방식이다. 보통 이미지 워핑을 구현하는 방법은 전방향 이미지 워핑과 역방향 이미지 워핑이 있다. 전방향 방법은 원본 이미지의 좌표를 기반으로 한 위치변환함수를 통하여 출력 이미지의 좌표를 구해내는 방법으로 출력 이미지에 매칭되는 점이 없는 홀이 생성되어 노이즈가 발생하는 단점을 가지고 있다. 이에 따라, 출력 이미지의 좌표를 기반으로

그림 2. 로봇 모델링.

Fig. 2. The robot modeling.

그림 3. (a) 로봇 시스템 (b) 어안 이미지.

Fig. 3. (a) The robot system (b) fisheye image.

그림 4. (a) 워핑 이미지, (b) 특징점, (c) Ego motion을 위한 움직임 벡터.

Fig. 4. (a) Warping image, (b) Feature points, (c) Motion vectors for ego motion.

그림 5. 원본 이미지의 위치.

Fig. 5. The position of original image.

원본 이미지의 좌표를 구하는 역방향 방법을 사용하며 블러링이 발생하지만 홀이 발생하지 않아서 노이즈가 없는 이미지를 획득할 수 있다. 식 (1)은 역방향 이미지 워핑 방법을 나타낸 것이다.

$$\begin{aligned} x_u &= f_x(x_d, y_d) \\ y_u &= f_y(x_d, y_d) \end{aligned} \quad (1)$$

여기서 x_u, y_u 는 원본 이미지의 (x, y) 좌표를 나타내며, f_x, f_y 는 각각 x, y 위치변환함수이다. x_d, y_d 는 출력이미지의 (x, y)

좌표를 나타낸다. 역방향 이미지 워핑 방법으로 원본 이미지의 위치를 얻으려면 먼저 그림 5에서 보는 것과 같이 r, θ 를 구하여야 한다. 그림 5는 화각이 185° 인 어안렌즈와 CCD 이미지 센서의 크기가 1/3으로 구성되어 있기 때문에 실제 이미지 정보는 원형으로 표현이 된다. 출력 이미지에서 가로축은 원본이미지의 각도, 세로축은 원본 이미지에서 중심점과의 거리로 대응하여 식 (2)에서 보는 것과 같이 출력이미지의 좌표를 이용하여 θ, r 을 구하고 이를 이용하여 식 (3)과 같이 원본 이미지의 좌표를 구하여 위치변환을 하게 된다.

$$\theta = x_d \frac{2\pi}{W_d}, \quad r = \frac{H_d}{2} - y_d \frac{H_u}{2H_d} \quad (2)$$

$$\begin{aligned} x_u &= x_c - r \cos \theta \\ y_u &= y_c - r \sin \theta \end{aligned} \quad (3)$$

여기서 W_d, H_d 는 출력이미지의 가로 세로 크기를 나타내며 그림 4와 같은 경우 2560과 480의 나타내게 된다. H_u 는 원본 이미지의 세로 크기를 나타내며 (x_c, y_c) 는 원본 이미지의 중심좌표를 나타낸다. 그림 3(b) 어안이미지를 이런 방식으로 변환하게 되면 그림 4(a)처럼 출력된다

4. LKOF을 이용한 Planar flow

4.1 Planar flow를 위한 특징점 배치

본 논문에서 사용하고 있는 이미지는 크기가 크고 전방향

에 대한 정보를 가지고 있기 때문에 이미지의 전체 영역에 대하여 움직임 벡터를 검출하여 이용하면 처리 시간이 길고 알고리즘의 신뢰성이 떨어지게 된다. 따라서 본 논문에서는 LKOF를 적용하기 위해 특징점들을 일정한 간격으로 배포하여 움직임 벡터들의 전체적인 방향성을 추출한다.

4.2 LKOF를 이용한 특징점 추출

Optical Flow는 연속하는 두 프레임 간 차이를 통해 움직임을 추정하는 방법 중 하나로서 이미지 프레임이 바뀌어도 픽셀의 밝기는 변하지 않고 객체의 움직임에 비해 시간의 변화가 더 빠르며 공간적으로 인접하는 점들은 동일한 움직임을 나타낸다고 가정한다. 이를 바탕으로 연속하는 두 프레임간에 발생하는 움직임의 방향과 속도를 벡터적으로 나타내는 방식이다. 연속하는 두 프레임간의 밝기의 변화가 없음을 통하여 두 프레임간에 발생하는 Optical Flow는 아래 식 (4)와 같이 나타낼 수 있다[6,7].

$$\begin{aligned} f(x, y, t) &= I(x, y, t) \\ &= I(x + dx, y + dy, t + dt) \end{aligned} \quad (4)$$

여기서 dx, dy, dt 는 두 프레임간 변화량을 나타낸다. 이를 x축과 y 축의 속도성분으로 표현하고 한 픽셀에 관하여 x, y에 대하여 표현이 가능한 밝기 강도의 n차 행렬 A에 최소자승법을 적용하면 식 (5)처럼 각축에 대한 V_x 와 V_y 를 추출할 수 있다.

$$\begin{aligned} A^T A v &= A^T b, \quad v = (A^T A)^{-1} A^T b \\ \begin{bmatrix} V_x \\ V_y \end{bmatrix} &= \begin{bmatrix} \sum I_{x_i}^2 & \sum I_{x_i} I_{y_i} \\ \sum I_{x_i} I_{y_i} & \sum I_{y_i}^2 \end{bmatrix}^{-1} \begin{bmatrix} -\sum I_{x_i} I_{t_i} \\ -\sum I_{y_i} I_{t_i} \end{bmatrix} \end{aligned} \quad (5)$$

식 (5)은 일반적인 Optical flow의 움직임 벡터를 나타내며 LKOF 방식은 일반적인 방식과 달리 특징점 개수를 적절히 조정하기 위하여 가우시안 필터를 추가하여 일반적인 움직임 추정의 엄청난 연산 부하를 개선하였다. 식 (5)에 가우시안 함수 W , ω_i 를 적용한 것이 LKOF의 속도 성분을 나타내는 식 (6)이다.

$$\begin{aligned} A^T W A v &= A^T W b, \quad v = (A^T W A)^{-1} A^T W b \\ \begin{bmatrix} V_x \\ V_y \end{bmatrix} &= \begin{bmatrix} \sum \omega_i I_{x_i}^2 & \sum \omega_i I_{x_i} I_{y_i} \\ \sum \omega_i I_{x_i} I_{y_i} & \sum \omega_i I_{y_i}^2 \end{bmatrix}^{-1} \begin{bmatrix} -\sum \omega_i I_{x_i} I_{t_i} \\ -\sum \omega_i I_{y_i} I_{t_i} \end{bmatrix} \end{aligned} \quad (6)$$

이런 LKOF를 워핑한 이미지에 적용을 하면 그림 4(b)를 얻을 수 있다. 그림 4(b)는 움직임 전후의 특징점을 표현하고 있으며 녹색 점은 움직임이 있기 전의 특징점을 파란색 점은 움직임 후의 특징점을 나타낸다. 이 특징점을 기반으로 움직임을 추정하여 로봇의 각도와 위치를 추정하게 된다.

5. Ego-motion using RANSAC

5.1 RANSAC을 이용한 FOE 도출

기존 Ego-motion은 이미지 상에 뿌려진 특징점 RANSAC은 1981년 발표되었으며 측정 노이즈가 심한 원본 데이터로부터 모델 변수를 예측하는 방법이다[8]. 측정된 원본 데이터 중에 포함되어 있는 거짓정보를 제거하기 위해서 수학적 모델을 지정하고 이 모델의 변수를 결정하는데 최소한의 데이터를 이용한다. 이런 과정을 반복적으로 진행하면서 데이터

의 집합을 확장시켜 가면서 최적의 해를 찾는다. 일반적인 통계 방식은 많은 데이터에서 유효하지 않은 데이터를 제거하면서 해를 찾는 과정에서 많은 데이터를 사용하기 때문에 연산량과 연산시간이 많이 소비되지만 RANSAC은 반복 횟수만 적절하게 조절하면 최적의 해를 구하는데 일반적인 방법보다는 효율적인 알고리즘이다. 기본적인 RANSAC은 추정(hypothesis)과 검사(verification)의 2단계를 가지고 있다. 추정 단계는 원본 데이터에서 N개의 샘플 데이터를 선택하고 모델 변수를 예측한다. 검사단계는 예측한 모델 변수가 맞는지 검사하고 유효 여부에 따라서 유효 데이터 집합을 조정한다.

본 논문에서는 그림 4(c)에 보는 것과 같이 LKOF로 추출한 방향 벡터들의 소실점을 RANSAC으로 구하여 FOE로 삼아서 로봇의 이동정보를 얻었다. LKOF를 이용하면 전후의 특징점을 얻을 수 있고 이 특징점을 연결하면 벡터가 된다. 이 벡터를 직선의 방정식으로 표현하면 식 (7)와 같다.

$$\begin{aligned} a_i x + b_i y &= c_i \\ \begin{bmatrix} a_0 & b_0 \\ \vdots & \vdots \\ a_n & b_n \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} &= \begin{bmatrix} c_0 \\ \vdots \\ c_n \end{bmatrix} \\ A x &= C \end{aligned} \quad (7)$$

식 (7)에 a, b, c 는 벡터를 표현하는 직선의 방정식의 변수를 나타내고 x 는 소실점을 나타낸다. A의 Moore-Penrose pseudo inverse 계산하여 정리하면 식 (8)과 같고 x 를 구할 수 있다.

$$\begin{aligned} A^+ &= (A^T A)^{-1} A^T \\ x &= A^+ C \end{aligned} \quad (8)$$

이렇게 구한 소실점 x 와 직선의 방정식이 유효한 지를 판단해야 한다. 그림 6에서 보는 것과 같이 소실점에서 벡터를 포함하는 직선에 수직인 선의 길이를 평가하여 소실점과 벡터의 유효성을 판단하게 된다. 선의 길이 r 을 구하는 과정은 식 (9)와 같다.

$$\begin{aligned} y &= ax + \beta \\ &= \left(-\frac{\cos\theta}{\sin\theta} \right) x + \left(\frac{r}{\sin\theta} \right) \\ r &= x \cos\theta + y \sin\theta \end{aligned} \quad (9)$$

그림 6. 소실점의 유효성 판단.

Fig. 6. The validation of vanishing point.

5.2 거리와 각도 추정을 위한 Ego-motion

벡터들이 향하는 점 즉 소실점은 전방향 이미지에서 FOE가 된다. 전방향 이미지는 x 축으로 2560의 픽셀의 데이터를 가지고 있고 이를 360° 로 변환하면 $0.14^\circ/\text{pixel}$ 을 얻을 수 있다. 소실점의 x 좌표를 각도로 환산하면 로봇의 회전각도를 얻을 수 있다.

이전에 구한 유효한 벡터의 크기를 기반으로 이동거리를 추정한다. 실제로 특징점이 이미지 상에서 이동한 거리이기 때문에 픽셀 값으로 출력이 되지만 비율은 동일하기 때문에 거리 값으로 환산이 가능하다. 그러나 유효하지 않은 벡터를 제거하는 과정이 반드시 필요하고 FOE와 동일 선상에 있는 벡터의 크기를 분포를 판단하여 거리를 추정하였다.

6. Localization

로봇의 위치 인식은 모델링을 기반으로 한 EKF를 통하여 진행한다. 로봇의 상태 벡터는 식 (10)과 같다.

$$\vec{x}_{(k)} = [x, y, \theta]^T, \quad \vec{u}_{(k)} = [v_l, v_r]^T \quad (10)$$

여기서 x, y 는 로봇의 위치를 나타내고 θ 는 방위각을 나타낸다. 입력 v_l, v_r 은 각 바퀴의 속도이며 이 속도는 512 펄스의 엔코더 데이터를 4배하여 얻게 된다. 로봇의 상태 방정식을 구하면 다음과 같다.

• Prediction model

$$\begin{aligned} \vec{x}_{(k+1)} &= f(\vec{x}_{(k)}) + w_{(k)} \\ &= \begin{bmatrix} x_{(k)} + v_{(k)} \Delta t \cos \theta_{(k)} \\ y_{(k)} + v_{(k)} \Delta t \sin \theta_{(k)} \\ \theta_{(k)} + \Delta t \frac{v_r - v_l}{L} \end{bmatrix} \end{aligned} \quad (11)$$

• Measurement model

$$\begin{aligned} \vec{z}_{(k+1)} &= h(\vec{x}_{(k+1)}, m_{(k)}) + v_{(k)} \\ &= \begin{bmatrix} r_{(k)} \\ \varphi_{(k)} \end{bmatrix} = \begin{bmatrix} d_{Ego-motion} \\ \theta_{Ego-motion} \end{bmatrix} \end{aligned} \quad (12)$$

식 (11)는 로봇의 모션 모델이며 이 함수는 이전 상태에서부터 다음 상태를 예측하는데 사용한다. 식 (12)는 예측된 상태에서부터 로봇의 위치를 계산하는데 사용된다. 그러나 f 와 h 는 공분산에 직접적으로 적용될 수는 없으며 대신 자코비안 행렬이 계산되어야 한다. 이 모델을 바탕으로 예측(prediction)과 보정(update)가 반복적으로 진행하면서 오차를 줄여가며 로봇의 좌표를 추정하게 된다. 본 논문에서는 측정 모델의 변수를 Ego-motion을 통하여 얻은 거리와 각도로 하여 예측된 로봇의 좌표를 추정하였다.

III. 실험

1. 실험 환경

본 논문에서 제안한 알고리즘을 검증하기 위하여 두 번으로 나누어 실험을 진행하였다. 정지 상태에서 특정 위치를 움직여서 로봇의 위치를 판단하는 실험과 로봇이 이동하는 동안에 추정한 위치와 GVLS (Global Vision Localization System) 상에서 측정한 위치를 비교하는 실험을 진행하였다. 로봇이 움직이는 동안에 위치를 측정하기 위하여 GVLS를 구축하였다.

(a)

(b)

그림 7. (a) 실험 환경 (b) 전역 이미지를 이용한 위치 추적 프로그램.

Fig. 7. (a) The experiment environment (b) The position tracking program using global vision system.

그림 7(a)에서 보는 것과 같이 로봇의 위치를 파악할 수 있도록 색깔 패치를 붙이고 카메라를 천장에 설치하였다. 이 카메라를 통한 이미지를 이용하여 로봇들의 위치를 추적하고 이를 바탕으로 실험 결과를 검증하였다. 또한 바닥은 특징점이 없고 색깔 패치 인식이 용이한 실험을 위해 무광택으로 도색이 되어 있는 220cm x 180cm 크기의 로봇축구 경기장을 이용하였다. 그림 7(b)에서 보는 것과 같이 전역 이미지를 이용한 위치 추적 프로그램은 카메라를 통하여 들어온 색깔 패치를 인식하여 로봇의 위치와 각도를 추정한다. 단위 시간

표 1. GVLS의 신뢰성 실험 결과.

Table 1. The experimental results of reliability in GVLS(cm).

no	Real Data		GVLS		Error	
	x	y	x	y	x	y
1	80	70	87.7	77.6	7.7	7.6
2	45	80	45.6	79.3	0.6	-0.7
3	135	70	130.4	66.3	-4.6	-3.7
4	170	90	174.7	92.2	4.7	2.2
5	120	85	116.5	79.6	-3.5	-5.4
6	160	50	163.7	55.6	3.7	5.6
7	70	80	77.6	79.3	7.6	-0.7
8	130	45	137.8	39.8	7.8	-5.2
9	70	85	63.4	79.3	-6.6	-5.7
10	130	45	133.6	49.2	3.6	4.2
평균(절대값)					5.1	4.1

그림 8. (a) 전진 5cm, (b) 후진 5cm, (c) 왼쪽으로 20°, (d) 오른쪽으로 20°.

Fig. 8. (a) forward 5cm, (b) backward 5cm, (c) 20° to the left, (d) 20° to the right.

마다 로봇의 위치를 텍스트로 저장하도록 테스트 프로그램이 구현하였다. 약 190cm 높이에 설치된 카메라는 640x480 이미지를 얻게 되고 이는 1pixel당 약 0.32cm로 추정한다. GVLS의 신뢰성을 판단하기 위해 총 10회의 임의의 좌표에서 로봇의 위치를 카메라로 읽고 실제 위치와 비교한 결과는 표 1과 같다. 표 1에서 보는 것과 같이 오차 크기의 평균은 X축이 5.1cm, Y축이 4.1cm로 나타났다.

2. 정지 상태에서 위치 추정

정지한 상태에서 초기 위치에서 얻은 이미지와 특정 위치로 옮긴 후 얻은 이미지를 비교하여 실제 이동거리와 각도의 오차를 구하여 성능을 평가하게 된다. 로봇이 직진만 했을 경우, 회전만 했을 경우에 대하여 실험 결과를 얻었다. 그림 8은 각각 상황에서 유효 벡터를 표시한 것이다. 전·후진의 경우에는 FOE는 원으로 표현하였다. 특히 그림 8(c), (d)에서 로봇이 회전했을 경우 Planar flow가 선명하게 나타나는 것을 확인할 수 있다.

정지 상태에서 4방향으로 실험을 통하여 얻은 결과는 표 2과 같다. 전·후진은 5cm씩 진행하고 회전은 20도씩 진행하였다. 제안된 알고리즘은 각도와 거리를 근소한 오차로 추정

표 2. 정지 상태에서의 실험 결과 비교.

Table 2. The comparison of the experimental results in static states.

	Reference		Result		Error	
	θ	cm	θ	cm	θ	cm
Front	0	5	3.515	5.562	-3.515	0.562
Back	0	5	4.078	5.316	-4.078	0.316
Left	20	0	22.746	0	-2.746	0
Right	20	0	22.910	0	-2.910	0

할 수 있음을 실험 결과를 통하여 확인했다. 그러나 직·후진을 할 때에는 360°에 대한 각도를 구하기 때문에 오차가 발생하였다. 기존 연구에서의 결과 X축 3.77cm, Y축 5.48cm 보다 낮은 X축 0.44cm, Y축 0.03cm를 얻었다[11,16].

3. 이동 상태에서 위치 추정

본 실험에서는 2cm/s의 속도로 직진하는 로봇의 위치를 제안된 알고리즘과 별개의 실험 시스템인 GVLS에서 얻어서 비교한다. 두 시스템에서 얻은 로봇의 이동경로는 그림 9와 같다. 그림 9에서 Real robot은 제안된 시스템이 탑재된 실제 로봇의 위치를 나타내며 Reference robot은 GVLS에서 추출한 로봇의 전역 위치를 나타낸다. 실제로 로봇을 직진 운동을

그림 9. 로봇의 이동 경로.

Fig. 9. The x-y-position of robot.

그림 10. 실제 로봇의 움직인 거리와 GVLS에서 측정된 거리 사이의 오차.

Fig. 10. The error of the real robot's distance and distance in GVLS.

그림 11. 실제 로봇의 각도와 GVLS에서 측정된 각도 사이의 오차.

Fig. 11. The error of the real robot's angle and angle in GVLS.

명령했지만 지면 상태나 로봇의 상태로 인한 외부 요인으로 인해 직진으로 이동하지 않은 결과를 얻었다. 그러나 제안한 알고리즘과 GVLS에서 얻은 참고자료를 비교해보면 유사한 경로로 움직였다는 것을 확인할 수 있다. 로봇이 이동한 거리를 비교해보면 그림 10과 같다. 로봇의 경로가 완전히 변한 부분에서 많은 오차가 발생했지만 그 부분을 제외하고는 $\pm 2\text{cm}$ 의 오차로 거리를 추정하고 있다. 그림 11은 로봇의 각도를 비교한 결과이다. 제안한 알고리즘이 전방향에 대한 각도를 추정하기 때문에 로봇의 각도가 크게 변할 경우 큰 오차를 발생하게 됨을 확인했으나 그 부분을 제외하고 $\pm 4^\circ$ 정도의 오차로 각도를 추정하였음을 확인하였다. 실제 로봇의 동작에서 제안한 알고리즘이 로봇의 위치를 추정하는 탁월한 효과가 있음을 확인할 수 있다.

IV. 결론

본 논문에서는 로봇의 상단부에 설치된 유리관 끝 부분에 어안렌즈가 부착되어 있고 바닥을 향해 있는 카메라에서 얻은 이미지에 워핑 처리를 하여 이미지를 복원하고 여기에 특징점을 고르게 분포한 Optical Flow를 기반으로 한 Ego-Motion을 이용하여 자기 위치를 인식하는 알고리즘을 제안하였다. 기존 다양한 센서를 통한 위치 인식 연구는 많이 발전하였지만 전방향 위치 인식은 전방향을 다 측정이 가능한 큰 효과가 있음에도 왜곡과 처리 속도 때문에 연구가 활발하지 않았다. 본 논문에서 제안하는 왜곡 보정이 등거리 보정은 되지 않지만 이를 Optical Flow를 이용한 상태벡터로 보완함으로써 왜곡으로 인한 위치 보정 효과의 하락을 방지하고 전방향 이미지의 데이터를 많이 활용할 수 있는 방향을 제시하였다. 움직임이 있는 이미지를 통한 실험과 실제 로봇에 적용한 실험에서 낮은 오차의 범위 안에서 위치 인식이 되는 것을 확인함으로써 제안한 알고리즘의 신뢰성을 입증하였다.

추후 연구를 통하여 다양한 움직임을 가지는 로봇에 대해 알고리즘을 적용하여 알고리즘을 개선할 예정이다. 또한 어안렌즈를 이용하는 멀티로봇에 적용하여 개별 로봇의 위치 인식과 이를 바탕으로 한 대형제어에 대한 연구를 진행할 예정이다.

REFERENCES

- [1] Y. Yagi, "Omni-directional sensing and its applications," *IEICE Transactions on Information and Systems*, vol. E82-D, no. 3, pp. 568-579, Mar. 1999.
- [2] T. E. Boulton, R. Micheals, X. Gao, P. Lewis, C. Power, W. Yin, and A. S. Erkan, "Frame-rate omnidirectional surveillance and tracking of camouflaged and occluded and targets," *Second IEEE Workshop on*, pp. 48-55, Jun. 1999.
- [3] V. N. Peri and S. K. Nayar, "Generation of perspective and panoramic video from omnidirectional video," *Proc. of DARPA Image Understanding Workshop*, New Orleans, May 1997.
- [4] K. Miyamoto, "Fish eye lens," *Journal of Optical Society of America*, vol. 54, pp. 1060-1061, Aug. 1964.
- [5] A. Basu, and S. Licardie, "Alternative models for fish-eye lenses," *Pattern Recognition Letters*, vol. 16, pp. 433-441, 1995.
- [6] J. Chang, "Digital image translational and rotational motion stabilization using optical flow technique," *IEEE Transactions on Consumer Electronics*, vol. 48, no. 1, pp. 108-115, Feb. 2002.

- [7] S. M. Kim, J. H. Lee, S. G. Roh, and S. Y. Park, "The study of pre-processing algorithm for improving efficiency of optical flow method on ultrasound image," *Journal of IEEK(in Korean)*, vol. 47, no. 5, pp. 24-32, Sep. 2010.
- [8] M. A. Fischler and R. C. Bolles, "Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography," *Communications of the ACM*, vol. 24, no. 6, pp. 381-395, Jun. 1981.
- [9] M. Irani, B. Rousso and S. Peleg, "Recovery of ego-motion using image stabilization," *Proc. of IEEE Computer Society Conference on*, pp. 454-460, 1994.
- [10] T. Gandhi, and M. Trivedi, "Parametric ego-motion estimation for vehicle surround analysis using an omnidirectional camera," *Machine Vision and Application*, vol. 16, no. 2, pp. 85-95, Feb. 2005.
- [11] J. R. Kim, M. S. Lim, and J. H. Lim, "Omni camera vision-based localization for mobile robots navigation using omnidirectional images," *Journal of Institute of Control, Robotics and Systems (in Korean)*, vol. 17, no. 3, pp. 206-210, Apr. 2011.
- [12] S. Y. Lee, "Use of optical flow information with three cameras for robot navigation," *Journal of Institute of Control, Robotics and Systems (in Korean)*, vol. 18, no. 2, pp. 110-117, Feb. 2012.
- [13] T. Y. Tian, C. Tomasi, and D. J. Heeger, "Comparison of approaches to egomotion computation," *Proc. of IEEE Computer Society Conference on*, pp. 315-320, 1996.
- [14] E. Menegatti, A. Pretto, A. Scarpa, and E. Pagello, "Omnidirectional vision scan matching for robot localization in dynamic environments," *IEEE Transactions on Robotics*, vol. 22, no. 3, pp. 523-535, Jun. 2006.
- [15] H. J. Sohn and B. K. Kim, "An efficient localization algorithm based on vector matching for mobile robots using laser range finders," *Journal of Intelligent and Robotic Systems Archive*, vol. 51, no. 4, pp. 461-488, Apr. 2008.
- [16] S. J. Yoon, W. S. Han, S. K. Min, and K. S. Roh, "Global localization of mobile robots using omni-directional images," *Transactions of the KSME(in Korean)*, vol. 31, no. 4, pp. 517-524, 2007.

최 윤 원

2009년 영남대학교 전기공학과(공학사).
2011년 동 대학원 전기공학과(공학석사).
2011년~2013년 동 대학원 로봇공학과 박사과정 수료. 2013년~현재 한국전자통신연구원. 관심분야는 Image Processing, Robotics, Omni-Directional Vision, Multi-robot.

최 경 식

2001년 영남대학교 전자공학과(공학사).
2008년 동 전자전기통신 교육대학원 석사. 2011년 동 대학원 전기공학과(공학박사). 2013년~현재 교육부 직업교육정책과 교육연구사. 관심분야는 Robotics, multiple robots, neural networks, SLAM

최 정 원

1995년 영남대학교 전기공학과(공학사).
2002년 동 대학원 전기공학과(공학박사). 2003년~2006년 ㈜STX 중공업. 2009년~현재 영남이공대학 전기자동차과 교수. 관심분야는 Control Theory.

이 석 규

1979년 서울대학교 전기공학과(공학사).
1981년 동 대학원 석사. 1990년 UCLA 제어공학과 박사. 1982년~현재 영남대학교 전기공학과 교수. 관심분야는 Control Theory, Robotics.