

# 어안 이미지 기반의 움직임 추정 기법을 이용한 전방향 영상 SLAM

## Omni-directional Vision SLAM using a Motion Estimation Method based on Fisheye Image

최 윤 원, 최 정 원, 대 염 염, 이 석 규\*  
(Yun Won Choi<sup>1</sup>, Jeong Won Choi<sup>2</sup>, Yanyan Dai<sup>3</sup>, and Suk Gyu Lee<sup>3,\*</sup>)

<sup>1</sup>Automotive IT Platform Research Team, ETRI

<sup>2</sup>Department of Automatic Electrical Engineering, Yeungnam College of Science & Technology

<sup>3</sup>Department of Electrical Engineering, Yeungnam University

**Abstract:** This paper proposes a novel mapping algorithm in Omni-directional Vision SLAM based on an obstacle's feature extraction using Lucas-Kanade Optical Flow motion detection and images obtained through fish-eye lenses mounted on robots. Omni-directional image sensors have distortion problems because they use a fish-eye lens or mirror, but it is possible in real time image processing for mobile robots because it measured all information around the robot at one time. In previous Omni-Directional Vision SLAM research, feature points in corrected fisheye images were used but the proposed algorithm corrected only the feature point of the obstacle. We obtained faster processing than previous systems through this process. 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 remove the feature points of the floor surface using a histogram filter, and label the candidates of the obstacle extracted. Third, we estimate the location of obstacles based on motion vectors using LKOF. Finally, it estimates the robot position using an Extended Kalman Filter based on the obstacle position obtained by LKOF and creates a map. We will confirm the reliability of the mapping algorithm using motion estimation based on fisheye images through the comparison between maps obtained using the proposed algorithm and real maps.

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

### 1. 서론

모든 로봇에게 자기의 위치와 주변 지도 작성의 방법은 로봇이 개발된 이후로 계속 되어온 로봇 연구자들의 고민 중에 하나이다. 로봇에 대한 연구가 시작된 이후 자기 위치 인식과 주변 지도 작성 즉, SLAM (Simultaneous Localization and Mapping)은 가장 핵심적인 연구 과제로서 많은 연구자들의 다양한 접근을 통한 연구를 바탕으로 EKF-SLAM, Fast SLAM 등의 다양한 알고리즘과 레이저 스캐너 센서, 카메라 센서, 초음파 센서 등 다양한 센서에 적합한 알고리즘이 개발되어 왔다. 최근 로봇 시스템은 컴퓨팅 성능을 보장한다면 충분히 적은 오차로 자기 위치와 지도를 작성할 수 있게 되었다. 그러나 컴퓨팅 성능을 확보하기 힘든 모바일 로봇이나 의료 로봇에 대한 SLAM 연구는 아직 미비한 점이 있다.

SLAM은 다양한 센서를 기반으로 연구되고 있는데 그 중에서도 레이저 스캐너 센서를 이용한 것과 카메라를 이용한 경우가 대표적이다. 기존에는 초음파나 IR 센서를 이용하여

특정 각도에 대한 장애물까지의 거리를 기반으로 하였으나 레이저 스캐너 센서가 활용이 되면서부터 분해능이 높아져서 세밀한 거리 정보를 바탕으로 주변 장애물 정보를 습득하게 되었다. 카메라를 이용한 경우는 사람이 대부분의 주변 정보를 시각을 통하여 얻는 것과 같이 로봇도 카메라를 이용하여 주변의 특징점 정보를 얻어 분석하여 자기 위치를 분석하게 된다. 카메라를 이용한 SLAM은 한 대를 사용하는 mono SLAM, 사람의 눈처럼 두 대를 사용하는 stereo SLAM, 실내 천장 정보를 이용하는 ceiling SLAM 등 다양한 대상으로 연구되었다. 이런 다양한 센서를 기반으로 얻은 주변 정보와 엔코더 정보를 바탕으로 한 odometry 정보를 변수로 적용한 Kalman filter에 대한 다양한 연구를 통해 SLAM의 정확도가 많이 상승하게 되었다. 그리고 최근에는 3D 레이저 센서나 RGB-D 카메라를 기반으로 한 3D SLAM에 대하여 많은 연구가 진행되고 있다.

특히 카메라를 이용한 다양한 형태의 VSLAM에서 단일 카메라를 이용한 mono SLAM 방식은 시스템이 단순하지만 단일 방향으로 정보를 얻기 때문에 주변 모든 정보를 획득하기에는 오래 걸리는 단점을 가지고 있으며 스테레오 카메라 방식은 동일 특징점을 카메라를 2대에서 측정하여 3차원 깊이 정보를 구하기 때문에 주변의 3차원적 정보를 얻는 장점을 가지지만 데이터 처리량이 많은 단점을 가지고 있었으나 최근에는 FPGA를 사용함으로써 많이 해결되고 있지만 이를 활용하여 SLAM을 하기에는 위험이 존재한다. 전방향 카메라

\* Corresponding Author

Manuscript received February 5, 2014 / revised May 20, 2014 / accepted June 3, 2014

최윤원: 한국전자통신연구원(yunwon.choi@etri.re.kr)

최정원: 영남이공대학교(choijw@ync.ac.kr)

대염염, 이석규: 영남대학교 전기공학과

(guangyanyan1129@hotmail.com/sglee@ynu.ac.kr)

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

방식은 어안렌즈나 거울을 이용하기 때문에 왜곡에 대한 문제를 가지고 있지만 360°의 주변 정보를 한 번에 얻기 때문에 처리 속도가 빠르고 많은 정보를 얻을 수가 있는 장점을 가지고 있다.

기존 전방향 이미지 SLAM는 포물면 반사경(paraboloidal mirror)에 반사된 이미지를 얻거나[1-4] 천장을 향해 설치한 어안 카메라로 전방향 이미지를 얻었다[5-7]. 그러나 이와 같은 방법은 카메라에 의해 가려지는 부분이 생기거나 조명에 영향을 많이 받는 단점이 있었다. 본 논문에서는 기존 방식과 달리 조명의 영향을 비교적 적게 받으면서 반사경의 왜곡 정보에 상관없는 주변 정보를 획득하도록 카메라를 바닥 방향으로 설치한 어안 렌즈를 이용하여 전방향 이미지를 획득하였다. 기존 연구에서는 위치 인식을 위하여 어안 렌즈의 왜곡을 보정하지 않고 그대로 이용하면서 이미지를 기반으로 한 자기 모션 추정을 통하여 자기 위치를 인식하는 알고리즘을 연구하였다. 기존 연구를 통하여 자기 위치는 인식하였으나 주변 장애물을 인식할 수가 없어 장애물을 인식하고 지도를 작성하는 지도작성에 대한 연구가 필요하게 되었다.

본 논문에서 전방향 이미지 SLAM의 지도작성 성능을 향상시키기 위하여 먼저 바닥을 향해 있는 어안 카메라를 통하여 얻은 전방향 이미지를 히스토그램으로 바닥부분에 대한 정보를 파악하게 된다. 바닥 부분을 제거하고 레이블링을 통하여 장애물에 대한 레이블을 지정하게 된다. 장애물의 모서리 점을 추출하고 이를 특징점 정보로 활용한 LKOF (Lucas-Kanade Optical Flow)를 이용하여 이동한 장애물의 특징점을 추출한다. 이렇게 연관 관계가 있는 특징 벡터의 시작점과 끝점만을 forward mapping을 통하여 좌표를 보정하고 특징 벡터의 각도와 크기 특징을 바탕으로 필터된 장애물 특징 정보를 서로 연결하여 평면에서의 장애물 지도를 작성하게 된다. 본 논문에서는 제안된 알고리즘을 실제 로봇에 적용하고 특정 경로를 움직이는 동안 생성된 지도 정보와 장애물의 실제 위치 정보를 비교하여 지도 작성 알고리즘의 성능을 검증하였다.

II. 제안한 시스템

1. 전체 구조

본 논문에서 제안한 알고리즘이 바닥 방향으로 부착된 어안 카메라를 이용하여 주변 정보를 한번에 획득하고 처리하여 지도를 작성한다. 그림 1에서 보는 순서도와 같이 로봇에 부착된 어안 카메라를 통하여 바닥 정보와 주변 정보가 포함된 이미지를 얻는다. 이미지를 분석하여 얻은 바닥면에 대한 히스토그램을 바탕으로 바닥면을 제거하고 장애물 후보에 대하여 라벨을 붙여서 나중에 보정된 특징점을 연결하기 위한 군집화의 정보로 활용하게 된다. 이와 동시 어안 이미지에서 모서리를 특징점으로 활용하여 움직임 벡터를 구하고 움직임 벡터의 크기와 각도의 분포를 바탕으로 불필요한 벡터를 제거한다. 유효한 움직임 벡터의 시작점과 끝점을 구형 모델링 정보를 바탕으로 보정하고 이 점은 움직이기 전후의 장애물 외관점으로 활용한다. 왜곡을 제거한 장애물의 외관점 정보와 장애물 라벨을 통한 군집화를 통하여 주변 지도를 작성하고 장애물의 위치를 추정한다.

2. 로봇 시스템

본 논문에서 제안한 알고리즘을 적용한 로봇 시스템은 그림

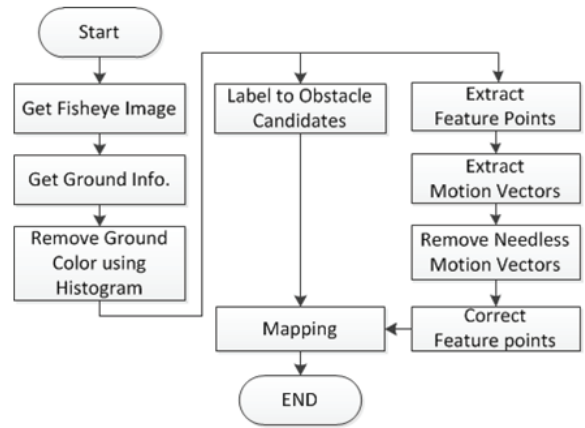


그림 1. 전체 시스템 블록도.  
Fig. 1. The system block diagram.

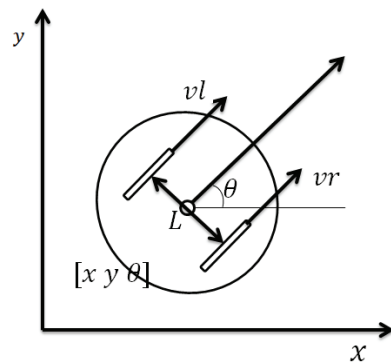


그림 2. 로봇 모델링.  
Fig. 2. The robot modeling.

2와 같은 모델링을 가지고 있으며 식 (1)과 같은 상태 벡터를 가지고 있다. 그림 2와 식 (1)에서 사용된  $x, y, \theta$ 는 좌표와 로봇의 각도를 나타내며  $L$ 은 바퀴간 거리,  $vr, vl$ 은 좌우 바퀴의 속도,  $v$ 는 로봇의 속도를 나타낸다. 이 모델링 정보를 바탕으로 구현된 실제 로봇 시스템은 그림 3(a)와 같으며 2개의 등근 바퀴를 가지고 있으며, 지름이 15cm이고 높이가 30cm인 원통형 소형 로봇이다. 이 로봇 가운데에 유리관을 세우고 어안렌즈가 부착된 카메라를 설치하여 주변 이미지를 얻게 된다. 어안 이미지는 그림 3(b)와 같으며 1280x960 크기의 이미지며 어안렌즈 화각이 185°이기 때문에 주변에 모든 이미지 정보가 한번에 저장된다.

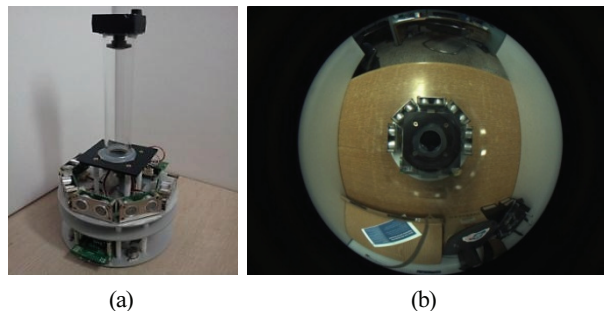


그림 3. (a) 로봇 시스템 (b) 어안 영상.  
Fig. 3. (a) The robot system (b) fisheye image.

$$\vec{x}_{(k)} = \begin{bmatrix} x_{(k+1)} \\ y_{(k+1)} \\ \theta_{(k+1)} \end{bmatrix} = \begin{bmatrix} x_{(k)} + v_{(k)}\Delta t \cos \theta_{(k)} \\ y_{(k)} + v_{(k)}\Delta t \sin \theta_{(k)} \\ \theta_{(k)} + \Delta t \frac{vr - vl}{L} \end{bmatrix} \quad (1)$$

3. 레이블링을 통한 장애물 후보 군집화

특징점을 추출하고 군집화를 할 경우 k-means, k-근접이웃, 인공신경망, 정규 베이지 분류기 등의 알고리즘을 보통 사용하지만 이 알고리즘들은 장애물의 수나 군집의 수가 다를 경우 큰 오류를 발생할 가능성을 가지고 있다. 본 논문에서는 기존에 바닥면의 정보를 제거하고 그 외의 부분을 레이블링을 통하여 라벨을 지정해둠으로써 군집화의 문제를 해결하였다. 그림 4는 어안 이미지에서 실제 영상을 나타내는 원형 부분의 이미지 정보를 히스토그램으로 나타낸 것으로 바닥면의 정보를 제거하고 레이블링 알고리즘을 적용한 결과는 그림 5와 같으며 사각형으로 표시된 영역을 라벨로 지정하였으며 이후 찾아진 특징점의 위치를 비교하여 라벨을 계속 가지고 처리를 하기 때문에 군집화의 성능을 향상시켰다.

4. LKOF을 이용한 장애물 특징점 검출

단일 카메라에서 얻은 이미지를 이용하여 랜드마크를 기반으로 자신의 위치를 추정하는 시스템은 이전 프레임과 현재 프레임을 비교하여 찾은 특징점을 기반으로 현재 위치를 추정하여야 한다. 그러나 다른 정보 없이 프레임 사이의 모서리와

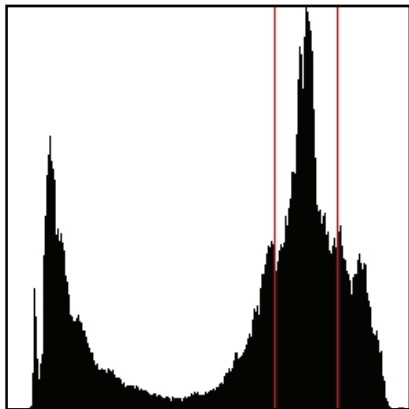


그림 4. 히스토그램을 이용한 바닥면 정보.  
Fig. 4. The information of floor surface using histogram.

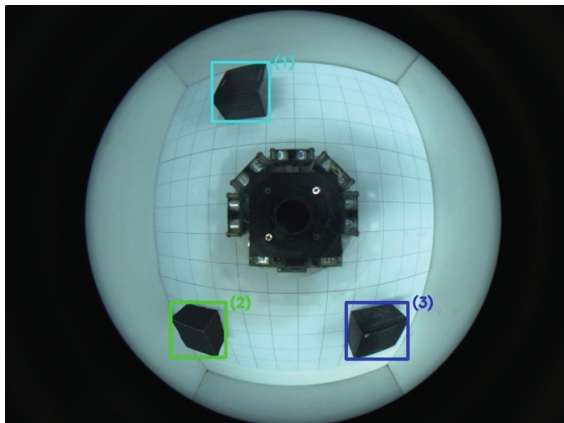


그림 5. 레이블링을 통한 장애물 후보 위치.  
Fig. 5. The position of obstacle candidates using labeling.

경계선 등의 특징을 비교하여도 상관관계를 쉽게 파악하기 어렵다. 그래서 본 논문에서는 optical flow를 이용하여 이전 프레임과 현재 프레임 사이의 특징점들의 관계를 추정하고 이 정보를 이용하여 로봇이 이동하기 전후의 랜드마크를 찾고 이를 활용하여 장애물의 위치를 추정하였다.

Optical flow는 연속하는 두 프레임 간 차이를 통해 움직임을 추정하는 기법으로 이미지 프레임이 바뀌어도 픽셀의 밝기는 변하지 않고 공간적으로 인접하는 점들은 동일한 움직임을 나타낸다는 가정하에서 연속하는 두 프레임 간에 발생하는 움직임의 방향과 속도를 벡터적으로 나타내는 방식이다. Optical flow는 모서리나 경계선을 추출하여 특징이 강한 영역에 대하여 움직임 벡터를 구해야 한다. 그래서 어안 이미지에서 아래 식 (2)와 같은 해리스 코너 검출을 이용하여 특징점을 추출한다[8].

$$C(x, y) = \begin{bmatrix} \sum_w (I_x(x_i, y_i))^2 & \sum_w I_x(x_i, y_i)I_y(x_i, y_i) \\ \sum_w I_x(x_i, y_i)I_y(x_i, y_i) & \sum_w (I_y(x_i, y_i))^2 \end{bmatrix} \quad (2)$$

$$O(x, y) = \det(C(x, y)) - k[\text{trace}(C(x, y))]^2$$

$$HCM(x, y) = \begin{cases} 1 & \text{if } O(x, y) > 0 \\ 0 & \text{otherwise} \end{cases}$$

$C(x, y)$ 는 가우시안 윈도우  $w$ 와 각 방향에 대한 밝기  $I_{x \text{ or } y}$ 를 바탕으로 밝기 변화를 나타내는 식으로 이를 이용하여 얻은 각 픽셀에 대한 모서리의 정도는  $O(x, y)$ 로 나타낼 수 있다. 모서리 정도를 일정한 값을 넘는 경우에만 이진화된 이미지인  $HCM(x, y)$ 를 얻을 수 있고 특히 해리스 코너 검출 알고리즘은  $k$ 를 0.04로 사용하여야 한다.

이렇게 얻은 모서리를 특징점으로 사용하여 optical flow를 적용하면 이전 프레임의 특징점을 시작점으로 하여 움직임 벡터의 방향과 크기를 얻을 수 있다. 위에 설명한 가정을 통하여 두 프레임간에 발생하는 optical flow는 아래 식 (3)와 같이 나타낼 수 있다[9].

$$f(x, y, t) = I(x, y, t) = I(x + \Delta x, y + \Delta y, t + \Delta t) \quad (3)$$

여기서  $\Delta x, \Delta y, \Delta t$ 는 두 프레임 사이에 이동한 정도를 나타낸다. 이를  $x$ 축과  $y$ 축의 속도성분으로 표현하고 모든 픽셀에 관하여 표현한 행렬에 최소자승법을 적용하면 식 (4)처럼 각 축에 대한  $V_x$ 와  $V_y$ 를 추출할 수 있다.

$$A^T A v = A^T b, \quad v = (A^T A)^{-1} A^T b$$

$$\begin{bmatrix} V_x \\ V_y \end{bmatrix} = \begin{bmatrix} \sum I_{x_i}^2 & \sum I_{x_i} I_{y_i} \\ \sum I_{x_i} I_{y_i} & \sum I_{y_i}^2 \end{bmatrix}^{-1} \begin{bmatrix} -\sum I_{x_i} I_{t_i} \\ -\sum I_{y_i} I_{t_i} \end{bmatrix} \quad (4)$$

식 (4)의  $A$ 는 인접한 픽셀의  $x, y$ 의 밝기에 대한 행렬을 나타내고,  $v$ 는  $x, y$  축의 속도 성분을 나타내고,  $b$ 는 시간에 따른 밝기의 변화 성분을 나타낸다. 일반적인 optical flow를 나타내는 식 (4)에 LKOF 방식은 Gaussian filter를 추가하여 특징점 개수를 적절히 조정함으로써 움직임 추정의 연산 부하를

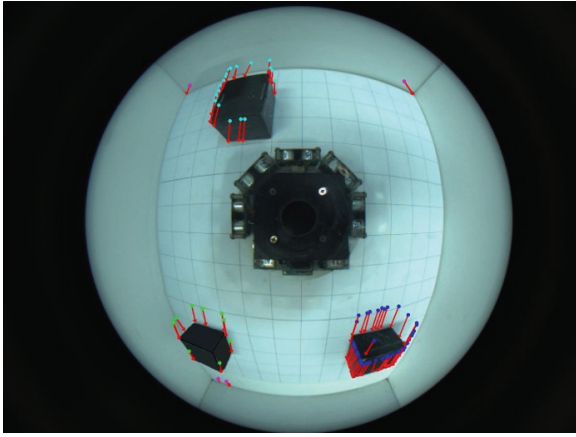


그림 6. LKOF를 이용한 움직임 벡터.  
Fig. 6. Motion vectors using LKOF.

개선하였다. 식 (4)에 가우시안 함수  $W$ ,  $\omega_i$  를 적용한 LKOF의 속도 성분( $V_x$ 와  $V_y$ )은 식 (5)에서 보는 것과 같이 도출이 가능하다.

$$A^T W A v = A^T W b, \quad v = (A^T W A)^{-1} A^T W b$$

$$\begin{bmatrix} V_x \\ V_y \end{bmatrix} = \begin{bmatrix} \sum \omega_i I_{x_i}^2 & \sum \omega_i I_{x_i} I_{y_i} \\ \sum \omega_i I_{x_i} I_{y_i} & \sum \omega_i I_{y_i}^2 \end{bmatrix}^{-1} \begin{bmatrix} -\sum \omega_i I_{x_i} I_{t_i} \\ -\sum \omega_i I_{y_i} I_{t_i} \end{bmatrix} \quad (5)$$

다양한 optical flow 중에서 Lucas-Kanade가 제안한 LKOF를 어안이미지에 적용한 결과는 그림 6에서 보는 것과 같으며 점은 움직이기 전의 특징점, 즉 시작점을 나타내며 화살표는 움직임 벡터의 크기와 방향을 이용하여 표시하였다. 상대적으로 로봇이 정면으로 움직였을 경우 반대 방향의 움직임 벡터 분포를 얻을 수 있다.

5. 왜곡 보정을 통한 랜드마크 보정 알고리즘

어안렌즈가 부착된 카메라를 통해 입력된 이미지는 어안 렌즈의 왜곡 때문에 왜곡되는 큰 단점을 가지고 있다. 기존 연구에서는 전체 좌표를 보정하여 보정된 이미지를 사용함으로써 왜곡 문제를 해결하려고 하였으나 어안 이미지를 보정할 경우 중심을 기준으로 확실한 영역에 한하여 보정하기 때문에 많은 픽셀 정보를 버리고 보정을 대상으로 하는 이미지의 크기가 한정적이며 각 픽셀마다 모두 연산해야 하기 때문에 속도도 느린 단점을 가지고 있다. 이에 본 논문에서는 특징점의 좌표만 보정을 통하여 원래의 위치를 찾고 나머지 픽셀을 보정하지 않는 방식으로 연산 속도를 개선하였다. 또한 보정을 대상으로 하는 이미지의 크기에 상관없는 연산 속도를 나타낼 수 있다. 그림 7은 어안렌즈의 투영모형을 나타내며  $F$ 는 초점 거리,  $(x_d, y_d)$ 는 왜곡 보정하기 전의 좌표,  $(x_u, y_u)$ 는 왜곡 보정 후의 좌표를 나타낸다[10]. 일반적으로 forward mapping은 이미지 매칭이 되지 않는 구멍이 발생하는 문제점 때문에 보간법과 같이 사용되지만 이미지가 흐려지는 문제점 때문에 대부분 backward mapping을 통하여 보정 후의 좌표에서 보정전의 좌표를 구하지만 본 논문에서는 로봇의 좌표만 추출하기 때문에 구멍이 발생하는 문제점이 없을 뿐 아니라 대상 이미지가 크기 때문에 forward mapping을 통하여 좌표를 추출하였다.

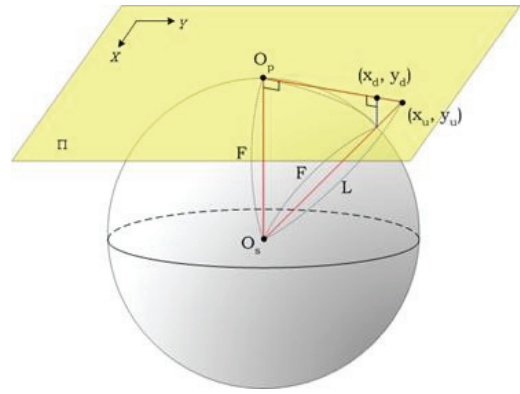


그림 7. 어안렌즈 투영 모델.  
Fig. 7. Fisheye projection model.

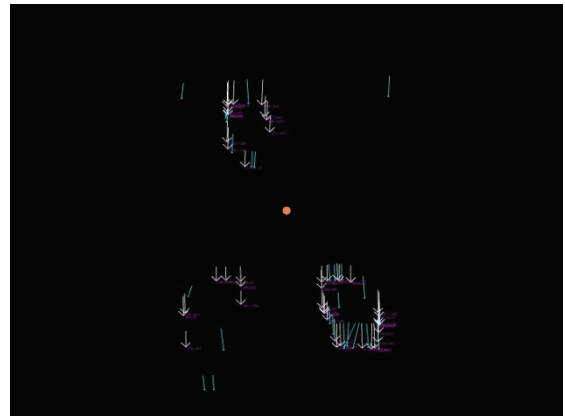


그림 8. 보정된 움직임 벡터.  
Fig. 8. The corrected motion vector.

$$(x_u, y_u) = \left( x_d \frac{L}{F}, y_d \frac{L}{F} \right) \quad (6)$$

$$L = \frac{F^2}{\sqrt{F^2 - r_d^2}} \quad (7)$$

식 (6)는 보정 전의 좌표를 이용하여 구한 보정 후의 좌표를 나타내며 식 (7)은 보정 전의 좌표 정보만을 이용하여 보정 후의 좌표를 구하는  $L$ 을 나타낸다. 여기서  $r_d$ 는 원점에서  $(x_d, y_d)$  좌표까지의 거리를 나타낸다.

어안 이미지에서 찾아서 보정한 특징점의 좌표는 그림 8과 같이 나타나며 보정이 되었지만 유효하지 않는 값을 가지는 벡터들을 제거하기 위해서 히스토그램을 이용하여 벡터의 크기와 각도를 기반으로 한 분포도를 구한다. 이 분포도의 패턴을 분석하여 그림 9와 같이 유효한 움직임 벡터를 추출하고 이 벡터의 시작점과 끝점 정보를 얻는다[11,12].

6. 장애물 특징점을 기반으로 한 지도 작성

필터 된 장애물의 특징점 집합의 외곽선을 계산하면 장애물을 추정할 수 있기 때문에 본 논문에서는 컨벡스 헐 알고리즘을 사용하여 특징점 집합의 외곽선을 계산하였다[13]. 컨벡스 헐 알고리즘은 이미지 상의 특정한 점 집합의 경계선을 추출하기 위한 알고리즘으로 개발되었으며 성질이 같은 점 집합의 외곽 정보를 계산할 수 있다. 컨벡스 헐은 점 집합을 입력으로 사용하며 집합 내에서 가장 아래 점을 기준으로

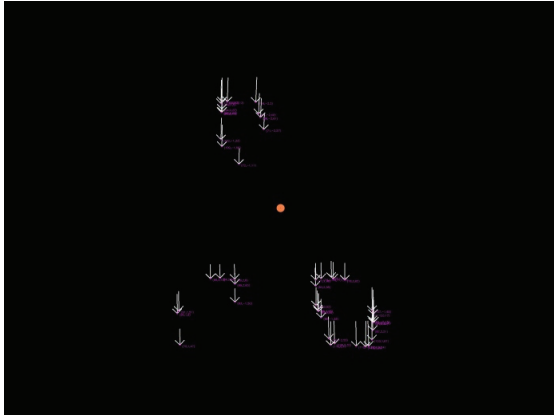


그림 9. 필터 된 장애물 특징점.  
Fig. 9. Feature points of filtered obstacles.

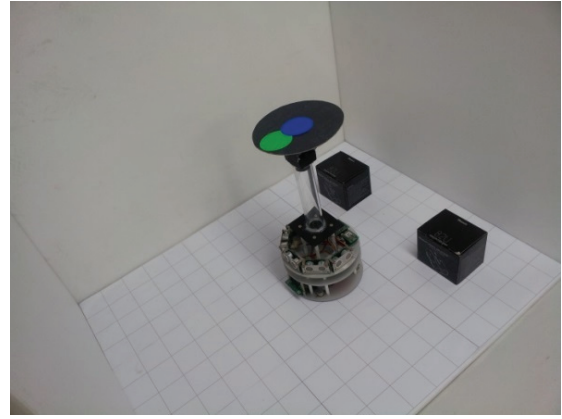


그림 11. 실험 환경.  
Fig. 11. The experiment environment.

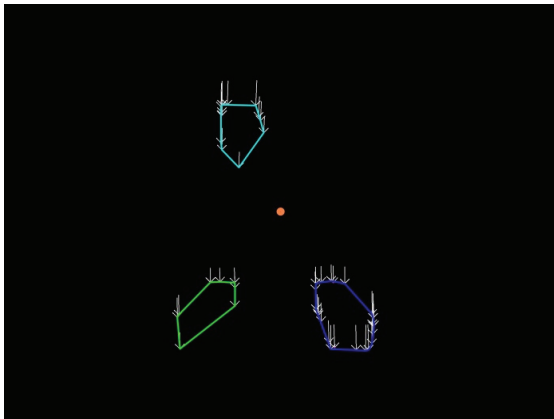


그림 10. 컨벡스 헐을 이용한 외곽점 연결.  
Fig. 10. The connection of outer points using convex hull.

하여 각도 별로 정렬한 뒤 순차적으로 세 점이 이루는 각도의 방향에 따라 제거와 포함 과정을 반복하여 시작점으로 되돌아 올 때까지의 최외곽점들의 정보를 산출한다.

본 논문에서는 이 컨벡스 헐 알고리즘을 사용하기 위해 미리 계산된 특징점들의 라벨 정보를 기반으로 점 집합의 성질 정보로 사용하여 라벨이 동일한 장애물에 대하여 외곽정보를 계산하여 그림 10과 같이 외곽점이 연결된 지도를 작성하였다.

### III. 실험

#### 1. 실험 환경

본 논문에서 제안한 알고리즘을 검증하기 위하여 다양한 장애물 환경에서 실험을 진행하였다[14]. 그림 11과 같이 바닥면이 5cm의 격자로 이루어진 실험환경에서 로봇보다 낮은 크기의 장애물을 설치하였다. 장애물의 위치는 전방향 이미지 SLAM의 알고리즘의 성능을 확인하기 위해 전방향에 대하여 장애물을 설치하고 실험을 진행하였다.

#### 2. 실험 결과

로봇의 중심으로부터 장애물의 실제 위치를 측정하고 제안된 알고리즘을 통하여 얻어진 장애물의 위치를 비교하여 실험 결과를 검증하였다. 표 1에서 보는 것과 같이 총 10회에 걸쳐서 장애물의 위치를 비교하였으며 실제 위치와 알고

표 1. 지도작성의 실험 결과 비교.

Table 1. The comparison of the experimental results in mapping.

	Reference		Proposed Algo.		Error	
	$\theta$	cm	$\theta$	cm	$\theta$	cm
1	-141.41	16.01	-140.06	28.00	-1.35	-11.99
2	-99.51	15.21	-107.28	26.64	7.77	-11.43
3	-40.62	23.05	-36.40	23.60	-4.22	-0.55
4	-19.99	29.26	-25.26	30.22	5.27	-0.96
5	0.00	27.50	0.00	27.78	0.00	-0.28
6	19.99	29.26	24.49	30.03	-4.50	-0.77
7	29.76	20.16	20.69	23.28	9.07	-3.12
8	38.62	23.05	36.40	23.60	2.22	-0.55
9	141.41	16.01	134.06	8.96	7.35	7.05
10	180.09	17.50	180.09	9.56	0.00	7.94
평균(RMS)					5.18	6.31

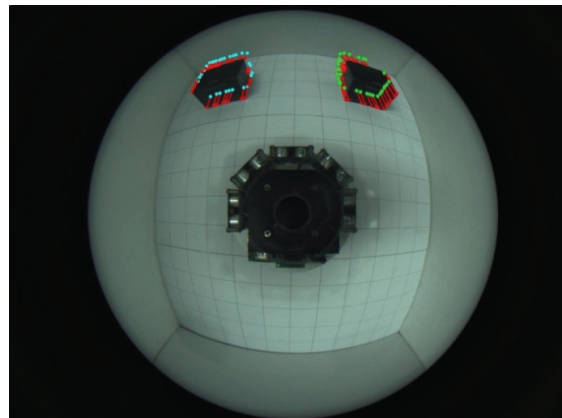


그림 12. 실험1의 움직임 벡터 검출.  
Fig. 12. Extraction of moving vectors in 1<sup>st</sup> experiment.

리즘을 통하여 얻어진 위치를 비교한 결과 오차의 평균으로 거리는 5.18 cm, 각도는 6.31° 를 얻을 수 있다[9]. 이 실험을 통하여 제안된 알고리즘은 로봇에 근접한 장애물에 대해서 오차가 크지만 대부분의 영역에서 작은 오차로 장애물의 위치를 찾는 성능을 가지고 있으며 또한 기존 연구에 비하여 렌즈에 대한 왜곡 보정만이 위치 추정에 영향을 주기 때문에 좀더 안정된 성능을 확보할 수 있음을 확인하였다.

그림 12와 14는 실험에서 움직임 벡터를 검출한 화면을

나타내고 있으며, 그림 13과 15는 지도 작성한 결과의 화면을 나타낸다. 빨간색 화살표는 왜곡된 영상에서의 움직임 벡터를 나타내고 흰색 화살표는 장애물의 특징점 벡터를 나타낸다. 실제 장애물의 모양은 동일하지만 제한된 알고리즘은 찾아낸 특징점의 위치에 따라서 지도에 나타나는 장애물의 모양은 달라지지만 위치는 유사하게 추정한다.

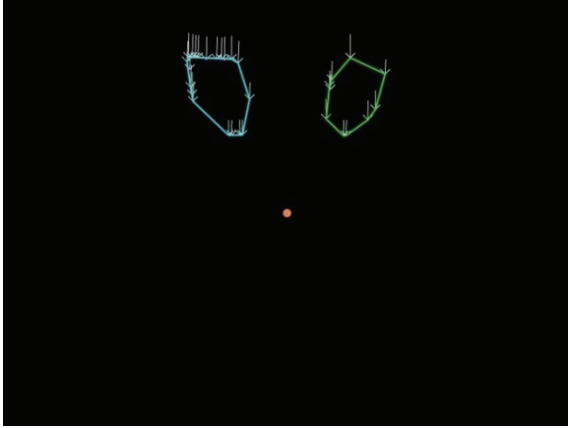


그림 13. 실험1의 지도 작성.

Fig. 13. Mapping in 1<sup>st</sup> experiment.

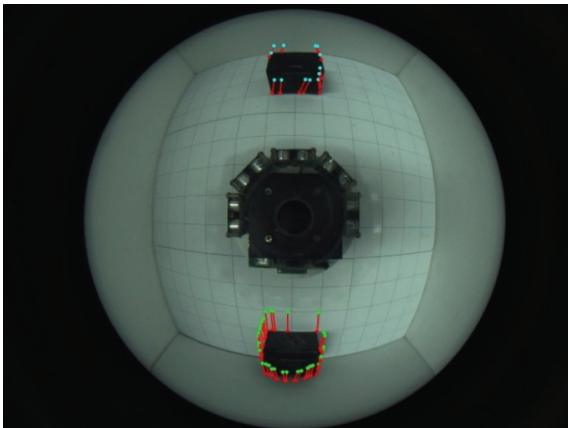


그림 14. 실험2의 움직임 벡터 검출.

Fig. 14. Extraction of moving vectors in 2<sup>nd</sup> experiment.

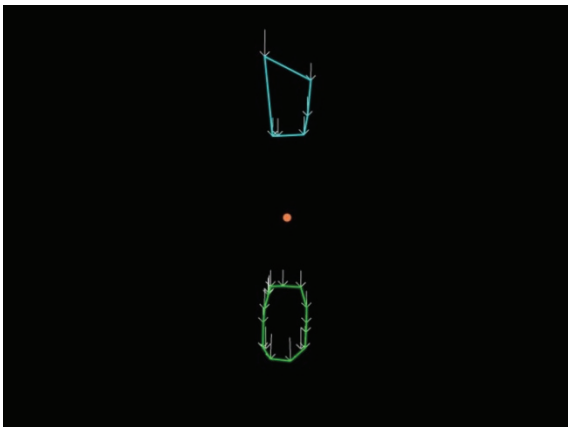


그림 15. 실험2의 지도 작성.

Fig. 15. Mapping in 2<sup>nd</sup> experiment.

#### IV. 결론

본 논문에서는 전방향으로 영상 정보를 얻을 수 있도록 어안카메라를 바닥면으로 향하여 부착한 로봇 시스템에서 지도를 작성하는 알고리즘을 제안하였다. 이 알고리즘은 어안 카메라를 통하여 들어온 전방향 이미지에 모서리를 추출하고 모서리의 움직임 벡터를 구하여 얻은 시작점과 끝점의 위치 정보를 보정한 후 장애물 특징점을 추출한다. 추출된 특징점 중에서 최외곽점을 연결하여 보정된 이미지 상에서 장애물의 모형을 만들어 지도를 작성한다.

최근 차량 주변 환경 인식에 어안 카메라가 많이 활용되면서 왜곡 보정에 관하여 많은 연구가 진행되고 있고 카메라 내부에 왜곡 보정이 된 영상을 출력하는 형태로 카메라가 발전하고 있기 때문에 한 번에 전체의 이미지 정보를 받을 수 있으며 지도 생성하는 시간도 적게 걸리면서 로봇 높이에 따른 장애물의 제약이 크게 줄일 수 있는 어안 이미지 기반의 전방향 이미지 SLAM에 대하여 앞으로 많은 연구해야 할 가치가 충분히 있다. 또한 지도 작성의 실험 결과에서 발견한 로봇에 근접한 장애물에 대한 큰 오차 문제를 추후 연구를 통하여 개선할 예정이며, 전방향 이미지 SLAM이 적용될 멀티로봇 지도 작성 알고리즘에 대한 연구를 진행할 예정이다.

#### REFERENCES

- [1] Y. Yagi, "Omni-directional sensing and its applications," *IEICE Trans, INF. & SYST.*, vol. E82-D, no. 3, pp. 568-579, Mar. 1999.
- [2] 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.
- [3] 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.
- [4] E. Menegatti, A. Pretto, A. Scarpa, and E. Pagello, "Omnidirectional vision scan matching for robot localization in dynamic environments," *Robotics, IEEE Transactions on*, vol. 22, no. 3, pp. 523-535, Jun. 2006.
- [5] T. Fukuda, S. Ito, F. Arai, Y. Yokoyama, Y. Abe, K. Tanaka, and Y. Tanaka, "Navigation system based on ceiling landmark recognition for autonomous mobile robot," *Proc. 1995 IEEE/RSJ International Conference on*, vol. 2, pp. 150-155, Aug. 1995.
- [6] J. Gaspar, N. Winters, and J. Santos-Victor, "Vision-based navigation and environmental representations with an omnidirectional camera," *Robotics and Automation, IEEE Transactions on*, vol. 16, no. 6, pp. 890-898, 2000.
- [7] 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.
- [8] W. J. Kim and C. G. Kim, "Text region extraction from videos using the Harris corner detector," *Journal of KIISE: Software and Applications*, vol. 34, no. 7, pp. 646-654, Jul. 2007.
- [9] 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*, vol. 47, no. 5, pp. 24-32, Sep. 2010.

- [10] J. H. Shin and G. J. Kwon, "Non-metric fish-eyeLens distortion correction using ellipsoid model," *HCI 2005 Conference on*, vol. 1, pp. 83-89, 2005.
- [11] 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.
- [12] 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.
- [13] R. L. Graham, "An efficient algorithm for determining the convex hull of a finite planar set," *Information Processing Letters*, vol. 1, pp. 132-133, 1972.
- [14] H. D. Kang and K. H. Jo, "Localization of 3D spatial information from single omni-directional image," *Journal of Institute of Control, Robotics and Systems (in Korean)*, vol. 12, no. 7, pp. 686-692, Jul. 2006.



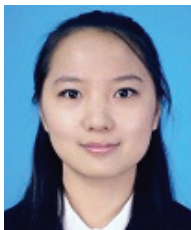
#### 최윤원

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



#### 최정원

1995년 영남대학교 전기공학과(공학사).  
2002년 동 대학원 전기공학과(공학박  
사). 2003년~2006년 (주)STX 중공업. 2009  
년~현재 영남이공대학 전기자동차과 교  
수. 관심분야는 Control Theory.



#### 대염염

2009년 Tianjin University of Science and  
Technology, China (공학사). 2011년 영남  
대학교 전기공학과. 현재 동대학원 박  
사과정, Multirobot formation, SLAM.



#### 이석규

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