

깊이 영상 기반 실내 공간 인식

김수경*, 최형일*

Indoor environment recognition based on depth image

Su-Kyung Kim*, Hyung-il Choi*

요약

본 논문에서는 실내 환경의 3차원 공간에서 벽면을 분리해내기 위해 깊이 카메라로 받아들인 영상을 이용한 방법을 제안한다. 논문의 실험 결과에서 얻을 수 있는 정보를 이용하면 실내 공간을 인식하거나 그에 따른 인접한 물체의 탐색 또는 벽면에 프로젝터를 투사하는 등 3차원 공간 활용에 용이하다. 논문에서 제안하는 방법은 먼저 3차원 입력 영상에서의 좌표 점들을 이용하여 법선 벡터를 검출하고, 검출된 법선 벡터를 비슷한 벡터들끼리의 그룹으로 나눈다. 나누어진 그룹들을 RANSAC을 이용하여 평면 단위로 분리한 후, 분리된 평면들은 실내 환경에서 알 수 있는 도메인 지식들에 기반 하여 벽면으로 구분 된다. 마지막으로 본 논문에서 제안하는 방법은 다양한 실험 환경을 통해 성능을 입증한다.

▶ Keywords : 법선벡터; RANSAC; 평면

Abstract

In this paper, we propose a method using an image received by the depth camera in order to separate the wall in a three-dimensional space indoor environment. Results of the paper may be used to provide valuable information on the three-dimensional space. For example, they may be used to recognize the indoor space, to detect adjacent objects, or to project a projector on the wall. The proposed method first detects a normal vector at each point by using the three dimensional coordinates of points. The normal vectors are then clustered into several groups according to similarity. The RANSAC algorithm is applied to separate out planes. The domain knowledge helps to determine the wall among planes in an indoor environment. This paper concludes with experimental results that show performance of the proposed

•제1저자 : 김수경 •교신저자 : 최형일

•투고일 : 2014. 9. 19, 심사일 : 2014. 9. 29, 게재확정일 : 2014. 10. 24.

* 숭실대학교 미디어학과 (Dept. of Media, Soongsil University)

※ 이 논문은 2014년 한국컴퓨터정보학회 제50차 하계학술대회에서 발표한 논문("깊이 영상에서 도메인지식을 이용한 실내 환경 인식")을 확장한 것임

※ 이 논문은 서울시 산학연 협력사업(SS110013)의 지원을 받아 수행된 연구임. 이 논문은 2013년도 정부(미래창조과학부)의 재원으로 한국연구재단의 지원을 받아 수행된 기초연구사업임(No.2013R1A1A2012012).

method in various experimental environment.

▶ Keywords : Normal vector; RANSAC; Plane

1. 서론

카메라로 받아들인 정보를 이용하여 영상내의 환경 인식을 하는 작업들은 이미 다양한 방법들을 통해 이루어지고 있다. 이를 위해 주로 전경과 배경을 분리하는 작업을 수행하기도 하며 여러 특징들을 이용한 물체 검출 작업이 이루어지기도 한다. 이와 같이 2차원 영상 또는 3차원 깊이 영상에서 배경과 물체를 분리해내는 것은 비전 분야에서 많이 다루어지고 있는 주제이기도 하지만 그만큼 어려운 문제이기도 하다.

기존의 물체 인식 방법들은 먼저 전경과 배경을 분리한 뒤에 전경을 검출하여 이용하는 것을 주로 목적으로 한다. 하지만 본 논문에서는 기존의 방법과는 다르게 입력된 3차원 실내 영상에서의 벽면만을 분리한 뒤 분리된 벽면 즉, 배경을 이용하는 것을 목적으로 한다.

벽면과 같은 실내 환경은 일반적으로 평면으로 구성되기 때문에 평면 검출을 이용한 방법으로 진행된다. 그러므로 본 논문에서 제안하는 방법은 전경과 배경의 구분 없이 전체 영상에서 평면들을 먼저 검출한 뒤 배경 벽면을 분리해 내는 작업으로 이루어진다.

기존에 많이 알려진 3차원 상에서의 평면 검출 방법으로는 Hough 변환[1,3], RANSAC[2], 벡터를 이용한 방법[4] 등이 있다. Hough 변환의 경우 주로 직선, 원, 타원 등을 검출하는데 이용된다. Hough 변환을 이용한 평면 검출 방법은 3차원 데이터를 이용하여 (x, y, z) 좌표를 Hough 변환 식에 의해서 (ρ, θ, φ) 좌표로 변환한 후 생성된 공간상에 나타나는 곡면들의 교점이 겹치는 부분을 찾아 평면을 검출하는 방법이다. 이 방법의 경우 모든 3차원 데이터에 대해 계산하기 때문에 처리속도가 느리고 메모리를 많이 차지한다는 단점이 있다.

RANSAC은 3차원 전체 데이터 중 임의의 샘플을 n 개 선택 후 파라미터를 결정한 뒤 모든 데이터와 비교하여 최적의 모델을 선택하는 것으로 이 방법 역시 한번 반복 수행 할 때마다 전체 데이터를 이용하여 처리속도가 오래 걸리는 문제가

있다.

본 논문에서는 속도 개선과 성능 향상을 위해 법선 벡터를 이용한 평면 검출 방법과 검출된 평면으로부터 벽면을 분리시키는 방법을 제안한다. 법선 벡터를 이용하여 인접 영역을 그룹화 한 뒤 RANSAC 알고리즘을 사용하게 될 경우 전체 데이터에 대한 계산이 필요하지 않고 그룹으로 나누어진 데이터에 대한 부분적인 계산만 필요하기 때문에 처리 시간이 감소된다. 검출된 평면으로부터의 벽면 검출은 도메인지식과 평면내의 정보들을 이용하는 방식으로 이루어진다.

본 논문은 그림 1과 같은 순서로 진행된다.

2장 본문의 2절에서는 3차원 좌표 점을 이용한 법선 벡터 검출 방법에 대해 정리하고, 3절에서는 검출된 벡터들을 레이블링 하는 과정에 대해 서술한다. 레이블링 된 데이터들을 그룹화하고 그룹으로부터 평면을 검출하는 과정은 4절과 5절에서 볼 수 있다. 마지막으로 벽면을 검출하는 과정에서는 실내 환경에서 얻을 수 있는 도메인 지식을 이용하는데, 일반적으로 실내 환경에서 알 수 있는 정보들은 1절에서 설명되었다. 3장에서는 실험 결과를 통한 성능 입증을 기술하며 4장에서는 결론과 향후 연구 방향에 대해 기술하였다.



그림 1. 시스템 흐름도
Fig. 1. The system flowchart

II. 본 론

1. 실내 환경에서의 조건

본 논문에서의 입력 영상에서 벽면은 그림 2에 음영처리된 부분처럼 1면에서 3면까지 검출이 가능하다. 영상에서 찾을 수 있는 깊이 값이 가장 큰 곳은 벽면임을 알 수 있다. 벽면과 벽면 사이는 90°의 각도로 이루어져 있다는 가정 하에 90°를 만족하는 평면을 검출한다.

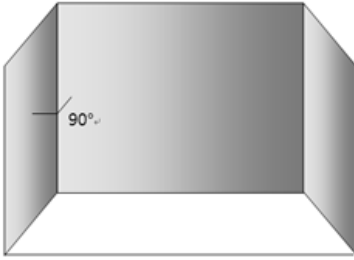


그림 2. 실내 공간
Fig. 2. indoor space

정리하면, 본 연구에서는 다음과 같은 작업 환경을 기반으로 하여 이루어진다. 검출되는 벽면은 중앙벽면을 기준으로 1면에서 왼쪽 벽면과 오른쪽 벽면 모두 총 3면까지 검출이 가능하며 모든 벽면은 곡면이 아닌 평면으로 이루어져야 한다. 또한 깊이 영상이기 때문에 카메라에서 가장 멀리 떨어져 있는, 즉 깊이 값이 가장 큰 부분이 기준 벽면이 된다. 벽면은 서로 떨어져 있지 않은 인접해 있는 실내 환경이며 서로 인접해 있는 벽면들 간의 각도는 90°로 이루어져 있어야 한다.

2. 법선 벡터의 검출

3차원 공간상에서 법선 벡터를 계산하기 위해 먼저 깊이 카메라에서 제공하는 영상의 좌표 점 정보들을 이용하여 법선 벡터를 구한다. 그림 3에서와 같이 하나의 3차원 좌표 점을 기준으로 하여 주위의 좌표 점들을 이용하여 계산한다.

그림 3의 점 P_i 를 중심으로 상하좌우에 위치한 점 4개를 이용하여 두개의 벡터 \vec{V}_1, \vec{V}_2 를 수식 (1)과 같이 계산한 뒤, 두 벡터를 외적 하여 법선벡터 \vec{V} 를 검출한다.

$$\vec{V}_1 = P_4 - P_2, \vec{V}_2 = P_3 - P_1 \quad (1)$$

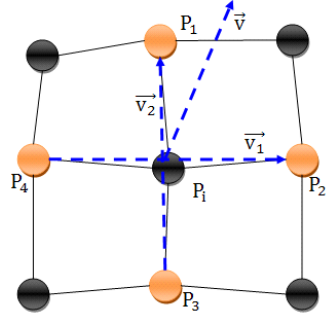


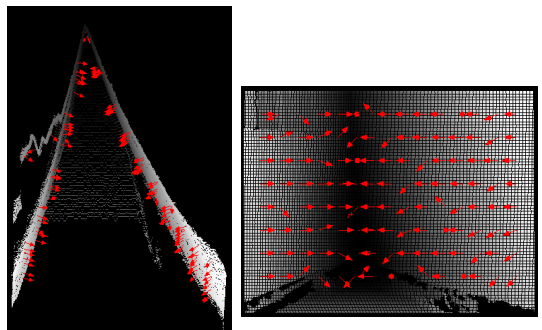
그림 3. 3차원 영상에서의 좌표 점과 벡터
Fig. 3. point and vector in three-dimensional image

두 벡터의 외적은 다음 수식으로 표현된다.

$$\vec{V}_1 = (a_x, a_y, a_z), \vec{V}_2 = (b_x, b_y, b_z) \quad (2)$$

$$\begin{aligned} \vec{V} &= \vec{V}_1 \times \vec{V}_2 = \begin{vmatrix} i & j & k \\ a_x & a_y & a_z \\ b_x & b_y & b_z \end{vmatrix} \\ &= (a_x, a_y, a_z) \times (b_x, b_y, b_z) \\ &= (a_y b_z - a_z b_y, a_z b_x - a_x b_z, a_x b_y - a_y b_x) \end{aligned} \quad (3)$$

단, $i = (1,0,0), j = (0,1,0), k = (0,0,1)$



(a) 법선 벡터 검출 (a) 위에서 내려다 본 영상
(b) 정면에서 바라본 영상
Fig. 4. detected normal vector (a) top view Image (b) front view Image

그림 4의 (a)는 3차원 영상을 회전시켜 위에서 바라본 모습이며 (b)는 정면에서 바라본 모습으로 평면에서의 법선벡터의 방향을 나타낸다. 검출된 법선 벡터는 이와 같이 벽면

에 수직으로 나타나는 것을 확인 할 수 있다.

3. 법선 벡터를 이용한 레이블링과 그룹화

2절의 과정에서 구해진 법선 벡터를 레이블링 하기위해 그림 5와 같은 Grassfire 알고리즘[5]을 이용한다. Grassfire 알고리즘은 기준이 되는 데이터와 값이 같거나 유사한 주변 데이터의 값을 바꾸는 방식으로 이루어진다. 주변 데이터 값을 바꿔 나가다가 유사한 데이터가 없을 경우 새로운 레이블을 생성하여 반복적으로 수행하며 더 이상 레이블을 할당 할 데이터가 없을 경우 종료 된다. 일반적으로 재귀함수를 이용하는 방법과 스택을 이용한 방법이 있는데 본 논문에서는 속도 개선을 위해 스택을 이용한 방식으로 구현하였다.

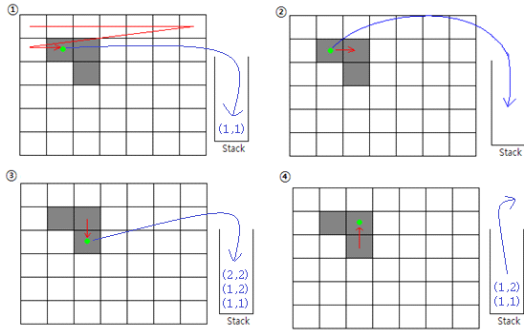


그림 5. Grassfire 알고리즘의 진행 순서
Fig. 5. operation of Grassfire algorithm

레이블링이 이루어지는 순서는 다음과 같다. 우선 전체 좌표 점의 레이블을 초기화한다. 초기화 후에 현재 좌표 점을 기준으로 주위 8방향 점들을 영상의 왼쪽에서 오른쪽, 위쪽에서 아래쪽을 기준으로 탐색한다. 기준에 만족하는 점이 있을 경우 현재 좌표 점과 동일한 레이블을 할당한 뒤 그림 5에서 보여 지는 것과 같이 Stack에 좌표 값을 넣어주고, 기준에 만족하는 점이 없을 경우 Stack에서 좌표 값을 삭제한다.

본 논문에서의 레이블링 기준은 두 법선 벡터 간의 각도 차이를 이용한다. 두 법선 벡터 사이의 각도는 내적을 이용하여 구한다.

두 벡터의 내적은 다음 수식으로 표현된다.

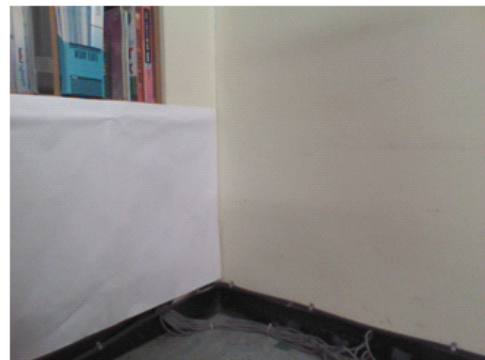
$$\begin{aligned} \vec{V}_1 \cdot \vec{V}_2 &= (a_x \times b_x, a_y \times b_y, a_z \times b_z) \\ &= |\vec{V}_1| |\vec{V}_2| \cos \theta \end{aligned} \quad (4)$$

$$\cos \theta = \frac{|\vec{V}_1 \cdot \vec{V}_2|}{|\vec{V}_1| |\vec{V}_2|} \quad (5)$$

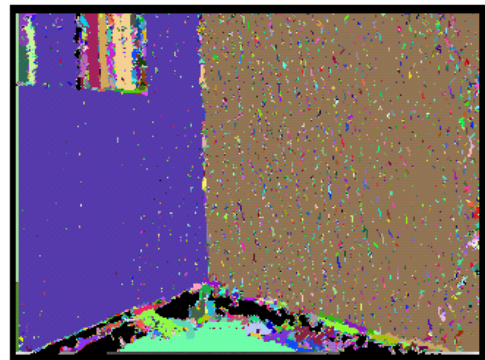
각도를 계산하기 위해 (5)번 수식에 역함수를 취해주면 다음 수식으로 나타낸다.

$$\theta = \cos^{-1} \frac{|\vec{V}_1 \cdot \vec{V}_2|}{|\vec{V}_1| |\vec{V}_2|} \quad (6)$$

다음으로 Stack의 top이 가리키는 좌표 점을 새 기준점으로 하여 반복 수행하는 방식으로 이루어진다. Stack이 비었을 경우 레이블이 할당되지 않은 좌표 점을 찾아 새 레이블을 할당한 뒤 앞의 과정을 반복한다. 모든 좌표 점에 레이블이 할당되었을 경우 알고리즘이 종료된다.



(a)



(b)

그림 6. 레이블링 결과 (a) 원본 영상 (b) 레이블링 영상
Fig. 6. Labeling result (a) Original image (b) Labeled image

레이블링이 종료된 후 동일한 레이블을 가진 좌표 점들을 각각의 그룹으로 생성한다. 이 과정을 통해 생성된 그룹들 중에 그림 6에서와 같이 잡음으로 분류되는 그룹들을 제외시키기 위해 표 1과 같이 그룹 내 좌표 점들의 개수가 많은 순서대로 그룹을 정렬한다.

4. RANSAC을 이용한 평면 검출

레이블링과 그룹화를 통해 정렬된 그룹들 내의 좌표 점의 개수들을 차례대로 합해 나가다가 구해진 합이 전체 영상 좌표 점 개수의 80% 이상이 되는 지점에서 멈춤 조건을 주고, 멈춤이 이루어지기 이전의 그룹들만을 이용하여 평면을 검출한다.

표 1. 내림차순 정렬된 레이블 그룹 예제
Table 1. Example of sorted descending Label group

| Label No. | Count |
|-----------|-------|
| 46 | 28913 |
| 42 | 22042 |
| 3640 | 5509 |
| 4326 | 1786 |
| 3501 | 1463 |
| 1 | 663 |
| 3856 | 639 |
| 11 | 625 |
| 19 | 476 |
| 4292 | 459 |

법선 벡터를 기준으로 이루어진 그룹들이기 때문에 평행한 여러 평면이 겹쳐져 있을 가능성이 있다. 또한 앞의 레이블링 과정에서 언급된 잡음으로 분류되는 그룹들 역시 제거하기 위해 평면들은 RANSAC plane fitting[2]을 이용하여 구분한다.

RANSAC 알고리즘은 입력된 데이터로부터 n개의 샘플을 선택한 후 최적의 파라미터를 예측한 뒤 비용을 측정하고 측정된 값이 만족스러운지 묻는 과정을 반복적으로 수행하며 가장 우수한 모델을 검출해 내는 것이다. 입력 데이터로부터 모델 검출까지의 과정은 그림 7과 같고 결과는 그림 8과 같다.

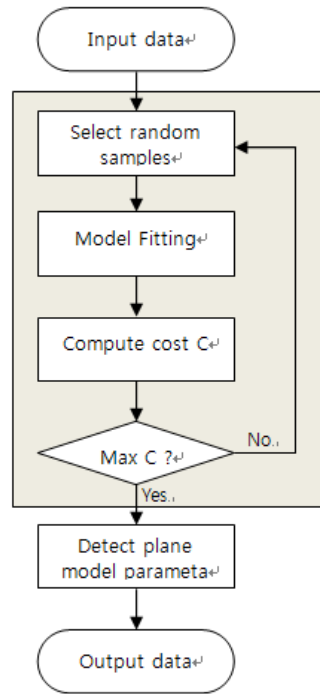


그림 7. RANSAC 흐름도
Fig. 7. RANSAC flowchart

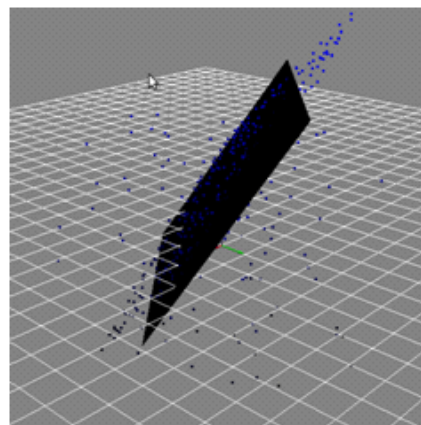


그림 8. RANSAC 평면 예측
Fig. 8. RANSAC plane fitting

RANSAC만을 이용하여 평면을 검출하는 경우 선택된 샘플로부터 얻어진 파라미터 모델과 입력된 모든 데이터에 대해 비교하는 연산이 이루어지므로 정확한 검출을 위해 반복 횟수가 늘어나게 된다. 또한 평면 영역의 검출은 파라미터 모델과 입력 데이터 간의 거리를 임계값으로 비용을 측정하기

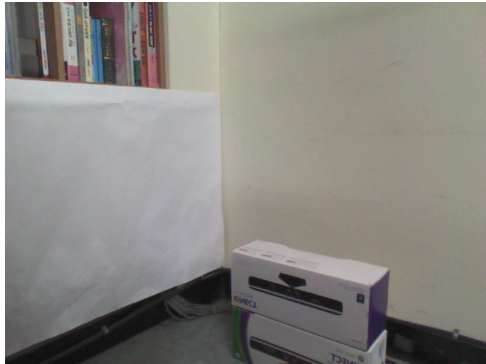
때문에 모델에 가까운 다른 평면 영역의 데이터 역시 동일한 평면으로 인식하여 포함한다. 본 논문에서는 이러한 문제점을 개선하기 위해 앞서 언급한 레이블링 과정을 거친 후 RANSAC을 실행한다.

먼저 n개의 그룹으로 나누어진 데이터들 중에서 가장 많은

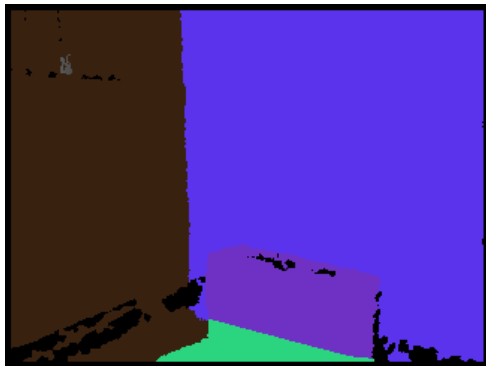
데이터를 포함한 그룹을 선택한다. 그룹 내의 데이터들 중에서 평면의 방정식을 이용하기 위해 임의의 3점을 샘플로 선택한다. 겹치지 않게 선택 된 3점의 샘플을 이용하여 평면 모델을 예측한 뒤 평면 성분을 검출한다. 검출 된 평면의 비용 값을 측정하고 비교하여 가장 높은 수치의 비용 값을 가진 모델을 선택한다. 최종 평면은 선택 된 모델을 구성하는 데이터들만을 이용하여 구한다.

본 논문에서의 영상 사이즈는 320*240으로 멈춤 조건이 발생하는 지점은 표 1에서의 8번째 그룹까지이며 따라서 총 8개의 평면을 검출하게 된다.

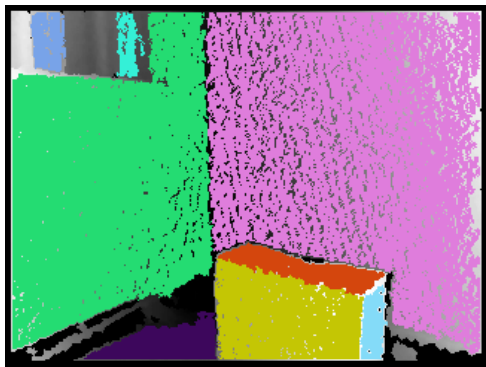
결과는 그림 9과 같다. (b)영상이 RANSAC만을 이용하여 평면을 검출한 결과 영상이다. 그림에서 보이는 것과 같이 벽면 앞의 물체영역은 하나의 평면으로 인식 되었고, 물체 중 일부는 바닥평면 기준과 더 가까워 바닥평면으로 인식 되었다. 반면 레이블링 과정을 거친 (c)영상에서는 가장 많은 데이터를 가진 두 벽면들이 깔끔하게 분리되어 검출 된 것을 볼 수 있고, 그 다음으로 평면성분 데이터가 많은 직육면체 물체의 3면과 바닥면 그리고 책장 부분의 평면 역시 잘 구분되어 검출되었음을 볼 수 있다.



(a)



(b)



(c)

그림 9. RANSAC 결과 (a) 원본 영상 (b) 일반 RANSAC 영상 (c) 레이블링 후의 RANSAC 영상
 Fig. 9. RANSAC result (a) Original image (b) General RANSAC result (c) RANSAC image after labeling

5. 검출 된 평면으로부터 벽면 분리

레이블링과 RANSAC을 이용해 검출된 평면들은 각 평면의 최고 깊이와 크기, 법선벡터, 평균 깊이, 좌 상단 좌표정보를 가지고 있다.

본론의 1절에서 설명했듯이 카메라로부터 가장 먼 곳은 벽면이기 때문에 우선적으로 평면들의 최고 깊이 값들을 비교하여 가장 큰 깊이 값을 가진 평면의 좌 상단 좌표가 영상의 중심을 기준으로 상단에 위치하는지 확인한다. 기준에 적합하지 않은 경우 두 번째 깊이 값을 가진 평면을 탐색하며 적합하다면 평면의 크기가 전체 영상 크기의 30% 이상인지 확인하고 맞을 경우 기준 벽면으로 인식한다. 만약 물체가 벽면을 가려 벽면의 크기가 30%가 되지 않을 경우를 고려하여 25% 이상인지 확인한 후 기준에 못 미칠 경우 벽면 이용에 적합하지 않다고 판단되어 평면으로만 검출될 뿐 벽면으로 인식되지 않는다.

검출된 벽면을 기준으로 수직인 평면들을 검색하여 선택된 평면들 중에 평면의 평균 깊이 값이 가장 크며 좌 상단 좌표가 영상의 상단에 위치하는 평면을 인접한 벽면으로 인식한다. 그림 10과 같이 두 평면 α_1, α_2 가 이루는 각도 θ 의 크기는 다음과 같이 계산한다.

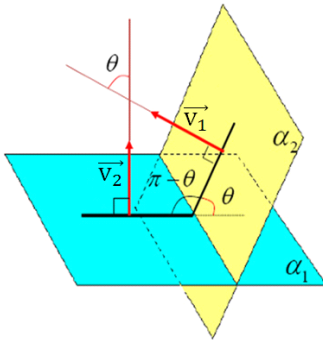


그림 10. 평면이 이루는 각도
Fig. 10. degree between adjacent planes

α_1, α_2 의 법선 벡터가 1.2절의 (2)번 수식과 같이 표현될 때, 평면 α_1, α_2 가 이루는 각도 θ 의 크기는 3절의 (5)번 수식과 같이 구해진다.

III. 실험 결과

본 논문에서 제안한 방법의 결과 확인을 위한 실험 환경은 다음과 같다. 실험에 사용된 영상 크기는 320x240 사이즈로 정규화 되었으며, 실험결과에 나오는 컬러영상과 깊이영상 모두 Microsoft사의 Kinect를 이용하여 실내 환경에서 촬영하였다.

먼저 4절의 그림 9의 결과에서처럼 전처리 없이 RANSAC을 실행하였을 경우 모델을 찾는 반복수행의 횟수가 30회 이상인 것에 비하여 레이블링 후 RANSAC을 이용할 경우는 반복수행 단 2번만으로도 평면을 검출할 수 있다. 전체 데이터를 이용하는 작업이기 때문에 실행속도 면에서 큰 개선이 이루어졌다. 또한 그룹 내의 데이터만을 이용하기 때문에 인접해 있는 다른 데이터에 영향을 미치지 않아 평면과 평면과의 사이 또는 평면과 물체와의 사이에서 나타나는 오류를 역시 감소하였다.

다음으로는 장애물이 없는 일반 벽면과, 장애물을 포함한 벽면에 그림 11, 12와 같이 본 논문에서 제안한 방법들을 순차적으로 적용시켜보았다.

그림 11의 (b)(c)는 (a)영상의 법선벡터 검출 영상이며 양쪽 벽의 벡터들은 그림과 같이 평면에 수직으로 나타난다. (d)는 레이블링을 마친 후의 영상이며 3차원 공간상의 이해를 돕기 위해 (e)와 같은 측면 영상으로도 확인하였다. (f)(g)는 RANSAC 과정까지의 영상이며 마지막으로 (h)(i)는 최종적으로 벽면만이 검출된 영상이다.

그림 11과 마찬가지로 그림 12에서와 같이 장애물을 포함한 영상 또한 동일한 과정을 통해 장애물을 제외한 두 벽면만을 찾아내었음을 확인 할 수 있다.

마지막으로 벽면을 분리해 내는 과정에서 오류를 확인하기 위해 여러 방식으로 벽면을 가려보았다. 벽면의 좌 상단 좌표 점이 영상의 상단부에 있다는 가정 하에 인식이 이루어지기 때문에 벽면의 상단 절반 이상이 가려져 평면의 좌 상단 좌표 점이 하단을 가리키는 경우 그림 11의 (b)에서와 같이 검출이 되지 않도록 하였다. 또한 벽면은 전체영상의 최소 15%를 만족시켜야 하므로 (d)처럼 15%를 만족시키지 못한 벽면은 인식하지 않도록 하였다.

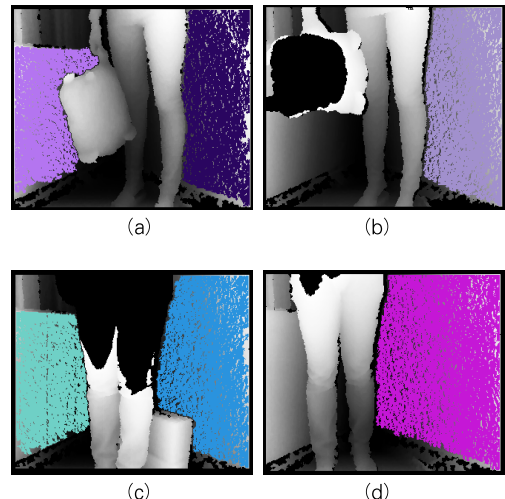


그림 11. 오류 검사 결과 (a) 일반 영상 (b) 벽면 상단 가림 영상 (c) 일반 영상 (d) 벽면 일부 가림 영상
Fig. 11. Error test Result (a) Normal image (b) upper occluded wall image (c) Normal image (d) portion occluded wall image

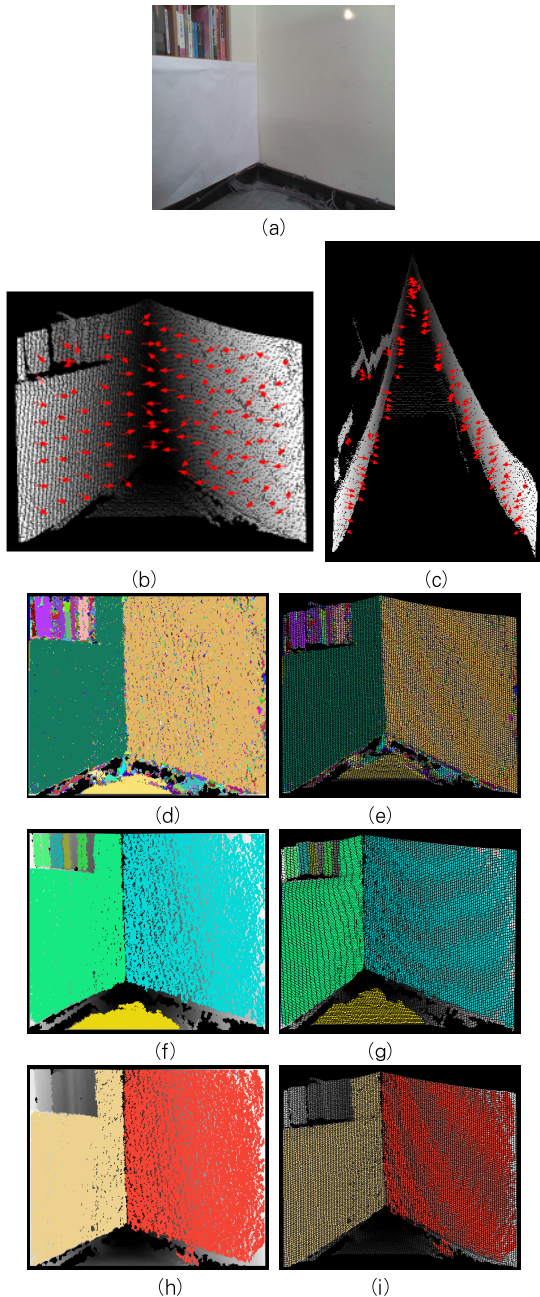


그림 12. 벽면 검출 (a) 원본 영상 (b) 법선벡터 측면 영상 (c) 법선벡터 윗면 영상 (d) 레이블링 영상 (e) 레이블링 측면 영상 (f) RANSAC 영상 (g) RANSAC 측면 영상 (h) 벽면 검출 결과 영상 (i) 벽면 검출 결과 측면 영상

Fig. 12. wall detection (a) original image (b) normal vectors (side view) (c) normal vectors (top view) (d) labeled image (front view) (e) labeled image (side view) (f) RANSAC image (front view) (g) RANSAC image (side view) (h) detected walls (front view) (i) detected walls (side view)

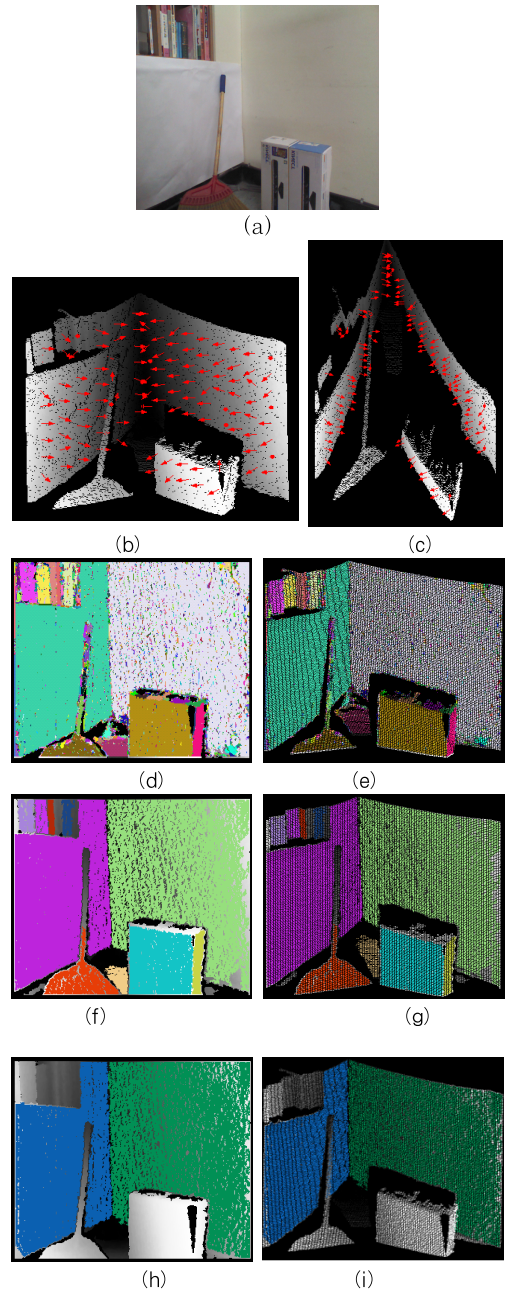


그림 13. 장애물을 포함한 벽면 검출 (a) 원본 영상 (b) 법선벡터 측면 영상 (c) 법선벡터 윗면 영상 (d) 레이블링 영상 (e) 레이블링 측면 영상 (f) RANSAC 영상 (g) RANSAC 측면 영상 (h) 벽면 검출 결과 영상 (i) 벽면 검출 결과 측면 영상

Fig. 13. wall detection when objects included (a) original image (b) normal vector (side view) (c) normal vector (top view) (d) labeled image (front view) (e) labeling image (side view) (f) RANSAC image (front view) (g) RANSAC image (side view) (h) detected walls (front view) (i) detected walls (side view)

IV. 결론

본 논문에서는 깊이 카메라로 받아들인 3차원 깊이 영상을 이용하여 검출된 평면들을 도메인 지식을 이용하여 물체를 제외한 벽면만을 검출하는 방법을 제안하였다. 향후 제안한 방법으로 검출된 벽면은 벽면의 위치를 찾아내어 프로젝터를 투사하거나 실내 환경 복원 등의 다양한 방법으로 활용 될 수 있을 것이다. 또는 벽면을 인식한다는 사실을 역으로 이용하여, 검출 된 벽면을 기준으로 벽면에 인접해 있는 물체들의 검출에도 응용 가능 할 것이다.

참고문헌

[1] Illingworth, John, and Josef Kittler. "A survey of the Hough transform." Computer vision, graphics, and image processing, Vol. 44, No. 1, pp. 87-116, Oct. 1988.

[2] Fischler, Martin A., and Robert 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.

[3] Borrmann, D., Elseberg, J., Lingemann, K., & Nüchter, A. "The 3D Hough Transform for plane detection in point clouds: A review and a new accumulator design", 3D Research, Vol. 2, No. 2, pp. 1-13, Jan. 2011.

[4] Dong-joong Kang, "A Stereo Camera Based Method of Plane Detection for Path Finding of Walking Robot", Journal of institute of control robotics and systems, Vol. 14, No. 3, pp. 236-241, Mar. 2008.

[5] Pitas, I. "Digital Image Processing Algorithms", Prentice Hall, 1993.

[6] Atid Shamaie and Alistair Sutherland, "A Dynamic Model for Real-Time Tracking of Hands in Bimanual Movements" GW 2003, LNAI 2915, pp. 172-179, 2004.

[7] RANSAC algorithm, <http://www.mrpt.org/>

tutorials/programming/maths-and-geometry/ransac-c-examples/

[8] Holz, D., Holzer, S., Rusu, R. B., & Behnke, S. "Real-time plane segmentation using RGB-D cameras" In RoboCup 2011: Robot Soccer World Cup XV, pp. 306-317, 2012.

[9] Sung-il Joo, Sun-hee Weon, Hyung-il Choi, "3D Pointing for Effective Hand Mouse in Depth Image", Journal of The Korea Society of Computer and Information, Vol. 19, No. 8, pp. 35-44, Aug. 2014.

저자 소개



김수경

2012: 한국산업기술대학교
컴퓨터학과 공학사.

현 재: 숭실대학교
미디어학과 석사과정

관심분야: 컴퓨터공학, 컴퓨터비전
Email : ksg1603@ssu.ac.kr



최형일

1979: 연세대학교 전자공학 공학사.
1982: 미시간대학교 전산학 공학석사.
1987: 미시간대학교 전산학 공학박사
1995: 퍼지 및 지능시스템학회 이사
1996: 정보과학회 컴퓨터비전 및
패턴인식 연구회 위원장

1997: IBM Waston Lab 방문연구원
2005: 한국정보과학회 이사
현 재: 숭실대학교 미디어학과 교수
관심분야: 컴퓨터공학
Email : hic@ssu.ac.kr