

6D ICP Based on Adaptive Sampling of Color Distribution

Eung-Su Kim[†] · Sung-In Choi^{††} · Soon-Yong Park^{†††}

ABSTRACT

3D registration is a computer vision technique of aligning multi-view range images with respect to a reference coordinate system. Various 3D registration algorithms have been introduced in the past few decades. Iterative Closest Point (ICP) is one of the widely used 3D registration algorithms, where various modifications are available nowadays. In the ICP-based algorithms, the closest points are considered as the corresponding points. However, this assumption fails to find matching points accurately when the initial pose between point clouds is not sufficiently close. In this paper, we propose a new method to solve this problem using the 6D distance (3D color space and 3D Euclidean distances). Moreover, a color segmentation-based adaptive sampling technique is used to reduce the computational time and improve the registration accuracy. Several experiments are performed to evaluate the proposed method. Experimental results show that the proposed method yields better performance compared to the conventional methods.

Keywords : Iterative Closest Point(ICP), 3D Registration, Color Segmentation, 6D Distance

색상분포에 기반한 적응형 샘플링 및 6차원 ICP

김응수[†] · 최성인^{††} · 박순용^{†††}

요약

3차원 정합이란 다시점에서 획득한 3차원 점군들을 정렬하는 기술로써 지난 수십 년간 많은 연구가 진행되고 있는 분야이다. 이러한 3차원 정합은 ICP(Iterative Closest Point) 알고리즘을 시작으로 많은 변형 ICP가 소개되고 있다. 하지만 ICP 계열의 알고리즘들은 최근접점을 대응점으로 간주하여 알고리즘을 수행한다. 그렇기 때문에 3차원 점군의 초기 오차가 큰 경우 정확한 대응점 탐색에 실패할 수 있다. 이런 문제점을 해결하기 위해 본 논문에서는 색상과 3차원 거리가 융합된 6차원 거리와 색상분포 유사도를 이용한다. 더 나아가 색상 분할 기반 적응형 샘플링을 이용하여 알고리즘 연산 속도를 감소시키고 성능을 향상시키는 것을 목표로 한다. 마지막으로 실험을 통해 기존의 방법과 본 논문에서 제안하는 방법의 성능을 비교한다.

키워드 : Iterative Closest Point(ICP), 3차원 정합, 색상 분할, 6차원 거리

1. 서론

국내외의 3차원 컴퓨터 비전 기술은 마이크로소프트사의 RGB-D(RGB-Depth) 센서인 키넥트가 시장에 소개된 이후로 이를 이용한 다양한 연구사례가 소개되고 있다[1-3]. 특히 무인이동로봇 또는 무인비행기의 위치 및 주변 환경지도를 실시간으로 생성하는 SLAM(Simultaneous Localization and Mapping) 관련 연구결과가 큰 주목을 받고 있다. 하지

만 RGB-D센서는 사람의 눈과 같이 물체의 가려진 부분을 볼 수 없기 때문에 다양한 방향에서 물체의 거리정보를 획득해야 물체의 온전한 형태의 3차원 정보를 획득할 수 있다. 이렇게 서로 다른 시점에서 획득된 거리정보를 공통된 좌표계로 이동시켜야 하는데, 이 기술을 3차원 정합(3D registration)이라고 한다.

3차원 정합은 Besl과 McKay[4]가 제안한 Iterative Closest Point(ICP)를 시작으로 다양한 변형 ICP[5-8]가 소개되고 있다. 그 중 Chen과 Medioni[7]는 점과 표면 정보를 이용하여 성능을 개선시킨 점대면(Point-to-Plane) ICP를 소개하였다. 최근에는 Segal[8]이 제안한 Generalized-ICP는 점대면 ICP의 성능을 개선하기 위해 두 개의 표면정보를 동시에 이용하였다. 하지만 ICP 계열의 알고리즘들은 최근접점을 대응점으로 간주하여 알고리즘을 수행하기 때문에 3차원 점군의 초기 오차가 큰 경우 정합에 실패하기도 한다.

※ 이 논문은 2014학년도 경북대학교 전임교원 연구년 교수 연구비에 의하여 연구되었음.

† 준회원: 경북대학교 컴퓨터학부 박사과정

†† 준회원: 삼성중공업 중앙연구소 책임연구원

††† 정회원: 경북대학교 컴퓨터학부 교수

Manuscript Received: October 22, 2015

First Revision: April 20, 2016

Accepted: May 9, 2016

* Corresponding Author: Soon-Yong Park(sypark@knu.ac.kr)

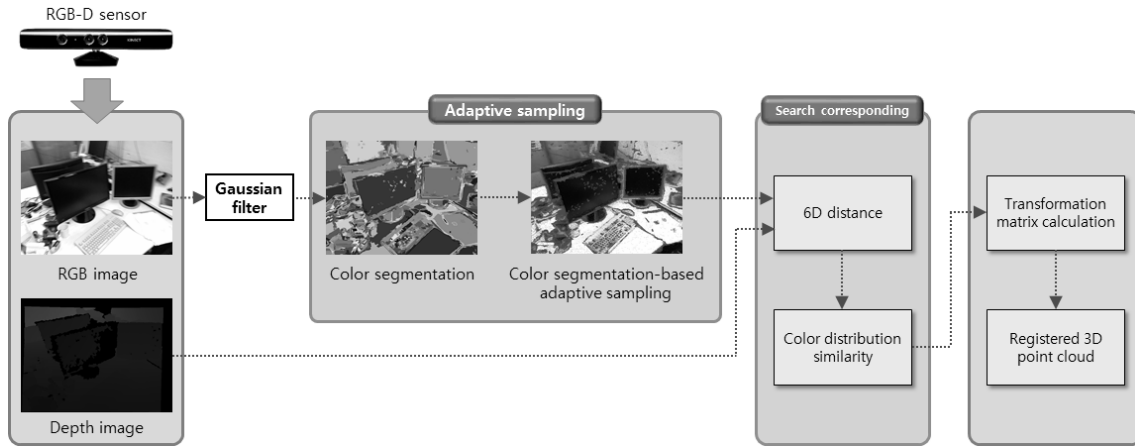


Fig. 1. Flow chart of the proposed method

이런 문제점을 보완하기 위해 색상정보를 이용하여 알고리즘의 성능 향상을 위한 연구가 진행되었다. 색상정보와 거리 정보를 융합한 형태의 다차원 거리를 이용하여 대응점 탐색을 통해 기존의 성능을 개선하려는 연구가 진행되었다[9-11]. 또 색상정보에 따라 3차원 점들을 정합에 용이한 형태 분류함으로써 정합 알고리즘의 성능을 향상시키는 연구가 진행된 적이 있고[12-13], 색상정보가 없는 3차원 점에 각 점들의 위치 정보를 이용하여 모의 색상(pseudo color)을 부여하고 이를 통해 성능을 개선하려는 연구사례도 보고된 바가 있다[14].

종전에 소개된 연구들을 통해 색상정보는 3차원 정보만 이용해서 해결할 수 없던 지역최소화(local minimum) 문제를 해결하는데 기여함을 알 수 있다. 하지만 이러한 방법은 점과 점간 색상정보만을 이용하기 때문에 색상이 같은 패턴으로 반복될 경우에는 알고리즘 성능이 저하되는 문제가 있다. 본 논문에서는 이러한 문제점을 해결하기 위해 대응점을 탐색할 때 6차원(6-dimensional) 거리와 색상분포 유사도를 이용한다. Fig. 1은 본 논문에서 제안하는 방법으로 요약하면 다음과 같다. 먼저 RGB-D 센서를 이용하여 RGB 영상과 깊이 영상을 획득한다. 이때 RGB 영상을 가우시안 필터(Gaussian filter)를 적용한 후, 이 영상을 여러 색상 영역으로 분할한다. 분할된 영역을 기반으로 적응형 샘플링을 적용하여 대응점 탐색에

유용한 샘플들을 획득한다. 이 샘플을 6차원 거리와 색상분포 유사도를 이용하여 최종적인 대응점을 선택함으로써 기존의 정합 성능을 개선한다.

2장에서 본 논문에서 제안하는 6차원 거리와 색상분포 유사도를 이용한 대응점 탐색 방법에 대해 소개한다. 3장에서는 2장에서 소개된 방법의 성능을 개선하기 위한 샘플링 방법에 대해 기술한다. 4장에서 실험을 통해 본 논문에 성능을 검증한다. 5장에서 결론 및 향후 연구 방향에 대해 기술한다.

2. 색상분포를 이용한 대응점 탐색

ICP 기반의 알고리즘들은 대응점을 탐색할 때 3차원 거리를 이용하고 비용함수(cost function)를 변형하여 성능을 개선한 연구들이 대부분이다. 하지만 3차원 거리만으로 찾은 대응점으로는 지역최소화 문제를 해결하는데 한계가 있다[15]. Johnson과 Kang[9]이 소개한 6차원 거리를 이용한 대응점 탐색은 점과 점간의 3차원 거리 및 색상 유사도를 비교하여 대응점을 탐색한다. 하지만 이러한 점대점 비교는 3차원 거리와 색상이 유사한 점이 다수 존재한다면 정확한 대응점 탐색에 어려움이 있다. 특히 Fig. 2a와 같이 3차원

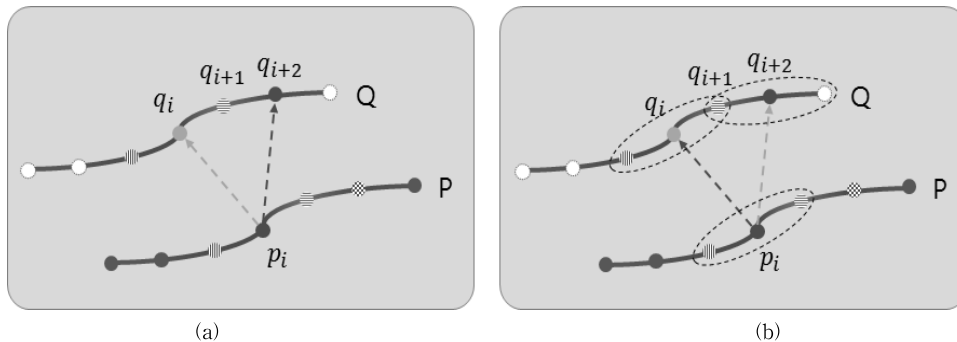


Fig. 2. Corresponding point search algorithm (a) Point-to-point algorithm (b) Color distribution-based similarity matching algorithm

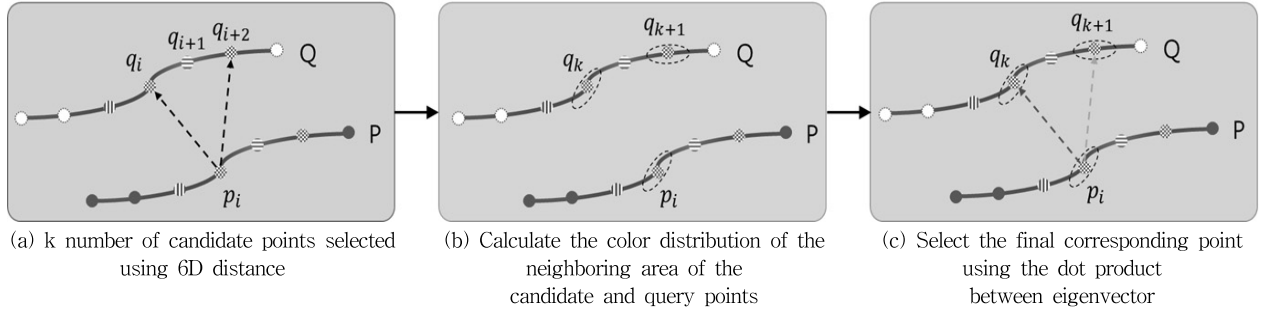


Fig. 3. Flow chart of the corresponding point search method using color distribution

원점 p_i 의 정확한 대응점은 q_i 이지만 센서의 오차로 인해 데이터가 정확하게 측정이 되지 않을 경우 다른 점을 대응점으로 선택할 수 있다. 이 문제를 해결하기 위해 본 논문은 Fig. 2b처럼 인접점들의 색상분포 유사도를 이용한 대응점 탐색 방법을 제안한다.

Fig. 3은 색상분포를 이용한 대응점 탐색 방법에 대한 흐름도이다. 간략하게 설명하면 다음과 같다. 먼저 6차원 거리를 이용하여 k 개의 대응점 후보를 탐색한다. 찾아진 대응점 후보(q_1, \dots, q_j)와 기준점(p_i)의 색상분포는 3차원 거리로 인접해 있는 점들을 이용하여 계산한다. 그리고 기준점의 색상분포와 가장 유사한 대응점 후보를 최종적인 대응점으로 간주한다. 2.1절에서 6차원 거리를 이용한 대응점 후보 탐색에 대해 설명하고 2.2절에서 색상분포 계산 및 비교 방법에 대해 자세하게 설명한다.

2.1 6차원 거리를 이용한 대응점 후보 탐색

3차원 점군 $P = \{p_i\}_{(i=1, \dots, n)}$ 와 $Q = \{q_i\}_{(i=1, \dots, n)}$ 가 있다고 가정한다. 이때 본 논문에서 제안하는 색상분포 유사도를 통한 대응점 탐색 방법을 사용하게 된다면, 기준점(p_i)의 대응점을 탐색하기 위해서는 점군 Q 의 모든 점들에 대해서 색상분포 유사도를 비교해야 된다. 때문에 대응점 탐색에 많은 시간이 소요하게 된다.

본 논문에서는 기준점으로부터 인접해 있는 점군 Q 의 k 개의 점들을 대응점 후보(q_1, \dots, q_k)로 간주하고, 이 점들에 대해서만 색상분포 유사도를 계산함으로써 대응점 탐색 시간을 단축시킨다. 이때 인접점은 Equation (1)에 정의된 6차원 거리를 이용함으로써 3차원 거리를 이용해 찾은 것보다 최종적인 대응점일 확률이 높은 점들을 선택할 수 있다.

$$d_6 = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2 + (z_1 - z_2)^2 + \alpha_1(r_1 - r_2)^2 + \alpha_2(g_1 - g_2)^2 + \alpha_3(b_1 - b_2)^2} \quad (1)$$

두 점의 위치는 $x_i, y_i, z_i (i=1,2)$ 이며 RGB 색상공간의 각 요소들을 $r_i, g_i, b_i (i=1,2)$ 로 나타낸다. $\alpha_i (i=1,2,3)$ 는 각 RGB 요소의 가중치이다.

2.2 색상분포 유사도

최종적인 대응점을 찾기 위해서 본 논문에서 기준점(p_i)과 번째 대응점 후보(q_k) 간의 색상분포를 비교한다. 먼저 색상분포 비교를 위해서는 인접점들의 색상분포 형태를 설명하는 공분산 행렬을 계산해야 한다. 이때 공분산 행렬 계산에 사용되는 인접점들은 3차원 거리로 인접해 있는 점들을 말한다.

p_i 의 m 개의 인접점들의 색상분포를 나타내는 공분산 행렬을 M_{p_i} , q_k 의 m 개의 인접점들의 색상분포를 나타내는 공분산 행렬을 M_{q_k} 라고 하며 각각의 공분산 행렬은 Equation (2), (3)처럼 계산한다.

$$M_{p_i} = \frac{1}{m} \sum_{p_j \in \Omega_p} (p_j - \bar{p}_i)(p_j - \bar{p}_i)^T \quad (2)$$

$$M_{q_j} = \frac{1}{m} \sum_{q_j \in \Omega_q} (q_j - \bar{q}_k)(q_j - \bar{q}_k)^T \quad (3)$$

이때 \bar{p}_i 는 p_i 인접점들의 중심 값이고, \bar{q}_k 는 q_k 인접점들의 중심 값이다. Ω_p 와 Ω_q 은 Fig. 4와 같이 각 p_i 와 q_k 의 m 개의 인접점들이 존재하는 공간을 의미한다.

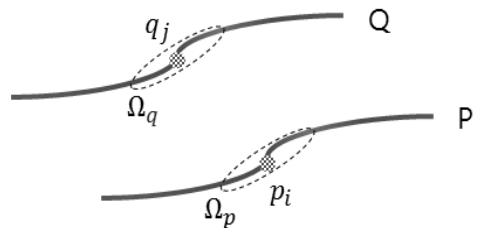


Fig. 4. Color distribution of the reference point and the candidate point generated using neighbor information

여기서 공분산 행렬 M_{p_i} , M_{q_j} 을 특이값 분해(singular value decomposition)를 통해 고유벡터와 고유값을 계산한다. 이때 계산된 고유값 하나가 다른 두 개보다 지나치게 높게 나온다면 이 대응점 후보를 외치점(outlier)으로 간주하고 제거한

다. 그 이유는 고유값 하나가 지나치게 높다는 것은 인접점들의 색상이 유사하다는 의미이기 때문에 색상분포의 유사도를 이용하여 대응점을 판단하기 어렵기 때문이다. 외치점이 제거된 대응점 후보와 기준점의 색상분포는 벡터연산을 통한 오차를 계산하기 위해 Equation (4), (5)처럼 M_{p_i} 의 고유벡터(E_1^p, E_2^p, E_3^p)와 M_{q_j} 의 고유벡터(E_1^q, E_2^q, E_3^q)를 사원수(quaternion)로 변환한다.

$$M_{p_i} \xrightarrow{SVD} (E_1^p, E_2^p, E_3^p) \longrightarrow qt(p_i) \quad (4)$$

$$M_{q_j} \xrightarrow{SVD} (E_1^q, E_2^q, E_3^q) \longrightarrow qt(q_j) \quad (5)$$

기준점과 각 대응점 후보간의 사원수를 Equation (6)과 같이 내적(dot product)을 통해 색상분포의 유사도를 계산한다. 이때 계산된 내적 값 d_c 가 1에 근접한 후보일수록 색상분포가 유사하다는 것을 의미한다. 대응점 후보 중 가장 높은 내적 값을 가진 한 점을 최종적인 대응점으로 간주한다.

$$d_c = qt(p_i) \cdot qt(q_j) \quad (6)$$

3. 색상 분할 기반 적응형 샘플링

2장에서 3차원 점군의 모든 점에 대해 인접점들의 색상분포를 계산하기 때문에 많은 계산 시간이 소요된다. 이 계산 시간을 감소시키기 위해 일반적으로 균등 샘플링(uniform sampling)을 사용할 수 있다. 하지만 Fig. 5a, 5b와 같이 올바른 정렬을 위한 특징 부분이 적은 경우 균등 샘플링을 사용하기에는 부적합하다. 만약 Fig. 5a처럼 균등 샘플링을 한다면 특징인 부분은 거의 선택하지 않게 되고 결국 올바른 정렬을 할 수 없다. Fig. 5b와 같이 특징인 부분에 대해서 조밀하게 샘플링을 하고 나머지 부분은 희소하게 샘플링함으로써 계산 시간 및 성능을 개선할 수 있다. 이와 유사한 방법으로 색상과 거리정보를 가진 RGB-D 영상에 적용할 수 있다.

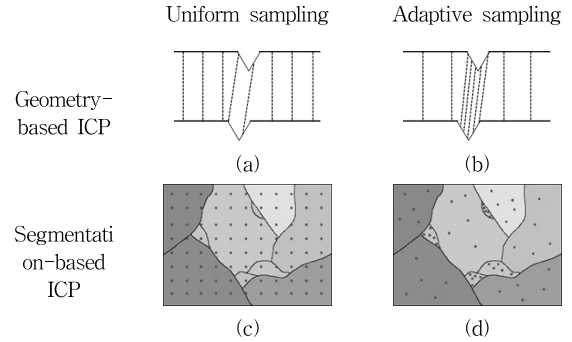


Fig. 5. Two different sampling method used for corresponding point search (a), (b) Geometry-based ICP (c), (d) Segmentation-based ICP

Fig. 6은 색상 분할 기반 적응형 샘플링 방법에 대한 흐름도이다. 먼저 RGB-D 영상의 색상을 그래프 기반 분할 방법 [16]을 이용하여 여러 색상 영역으로 나눈다. 이 방법은 $O(n \log n)$ 의 시간 복잡도와 안정적인 성능으로 널리 사용되고 있다. 또한 Fig. 7처럼 임계값(threshold)을 조절하여 분할할 영역의 크기를 조절할 수 있다. 이 임계값이 감소할수록 색상 영역을 더 많이 분할하고 임계값이 증가할수록 색상 영역은 더 적게 분할된다. 이때 샘플링에 가장 적합한 크기로 분할할 임계값은 5장에서 기술할 것처럼 실험적으로 알아낸다. 이렇게 여러 영역으로 분할된 RGB-D 영상을 Fig. 5c와 같이 균등 샘플링을 하지 않고 Fig. 5d처럼 각 영역별로 동일한 수의 샘플을 무작위로 선택한다. 색상 영상에서 특징이라고 할 수 있는 작은 영역에서는 조밀하게 샘플링하고 넓은 영역에서는 희소하게 샘플링한다. 결과적으로 특징인 부분을 많이 샘플링을 함으로써 성능은 향상되고 전반적으로 샘플수가 줄어들어 속도도 개선된다.

4. 실험 결과 및 분석

본 논문에서 제안된 방법의 성능을 검증하기 위해 Freiburg 데이터[17]를 사용했다. 실험은 최근접 이웃법, 6차원 거리, 균등 샘플링을 이용한 제안된 방법, 색상 분할 기반 적응형 샘플링을 이용한 제안된 방법 총 4가지의 대응점

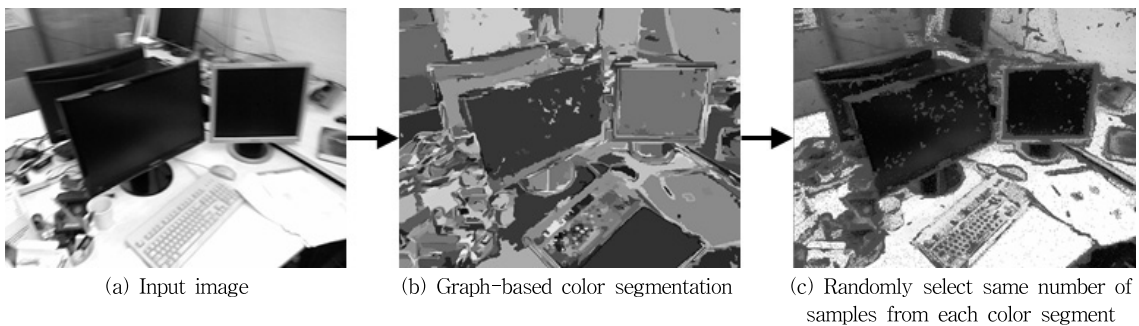


Fig. 6. Flow chart of the color segmentation-based adaptive sampling

data set	RGB image		
FR1 desk			
FR1 360			
FR1 floor			

Fig. 7. Sample RGB images of each data set

탐색 방법으로 나누어 시행하였다. 이 4가지의 방법들은 점대면 ICP의 비용함수 최적화 방법을 사용했다. 실험에 사용된 하드웨어는 Intel Core i5-4670 3.40GHz CPU, 8GB RAM, nVIDIA GeForce GTX 650 Ti로 구성하였다.

4.1 최적의 샘플링 파라미터 추정

Fig. 7은 실험에 사용된 데이터 세트의 RGB 영상이다. FR1 desk 데이터는 일반적인 사무 환경에서 주로 책상 위를 획득한 영상이다. FR1 360 데이터는 사무환경을 제자리에서 360도 회전하면서 획득한 영상이고 FR floor 데이터는 3차원 특징이나 색상의 변화가 거의 없는 바닥을 영상을 획득한 데이터이다. 실험에 사용된 그래프 기반 분할은 [18]에서 논문의 저자가 제공하는 소스를 이용한다.

먼저 최적의 그래프 기반 분할 적응형 샘플링을 위한 파라미터 알아내야 한다. Fig. 8에서 보이듯 그래프 기반 분할 임계값에 따라 분할할 영역의 개수를 조절할 수 있다. 또 각 영역에서 무작위로 선택할 점들의 수에 따라 결과 값이 달라질 수 있다. Table 1을 보면 영역을 많이 분할 경우, 즉 임계값이 작을 경우 전반적으로 성능이 향상되지만 계산 속도가 많이 늘어나는 걸 볼 수 있다. 하지만 임계값이 50으로 설정한 실험결과와 0으로 설정한 실험결과를 비교해보면 회전, 이동오차에서 큰 변화가 없지만, 50으로 설정한 실험이 계산 속도에서 이점이 있음을 보인다. 또 각 영역에서 무작위로 선택할 샘플의 수가 증가할수록 성능이 향상되는 경향을 보이지만 샘플의 수가 1000개 이후로는 차이가 거의 없다는 것을 확인할 수 있다. 결론적으로 실험에 사용될 파라미터는 나눌 영역 수를 결정하는 임계값을 50, 각 영역에서 선택할 샘플 수는 1000개가 가장 최적의 설정값임을 알 수 있다. 이렇게 찾아진 최적의 파라미터를 설정한 후 실험하였다. 이때 Table 1의 회전 오차는 Equation (7)를 이용하여 계산하고, 이동 오차는 Equation (8)을 이용하여 계산한다.

$$E_r = \arccos\left(\frac{\text{trace}(\Delta R)}{2}\right) \quad (7)$$

$$E_t = \|t_e - t_g\| \quad (8)$$

여기서 groundtruth의 회전변환행렬을 R_g , 각 알고리즘으로 추정된 자세의 회전변환행렬을 R_e 라고 할 때, 이들의 상대 회전변환행렬은 $\Delta R = R_e R_g^{-1}$ 이다. groundtruth의 이동

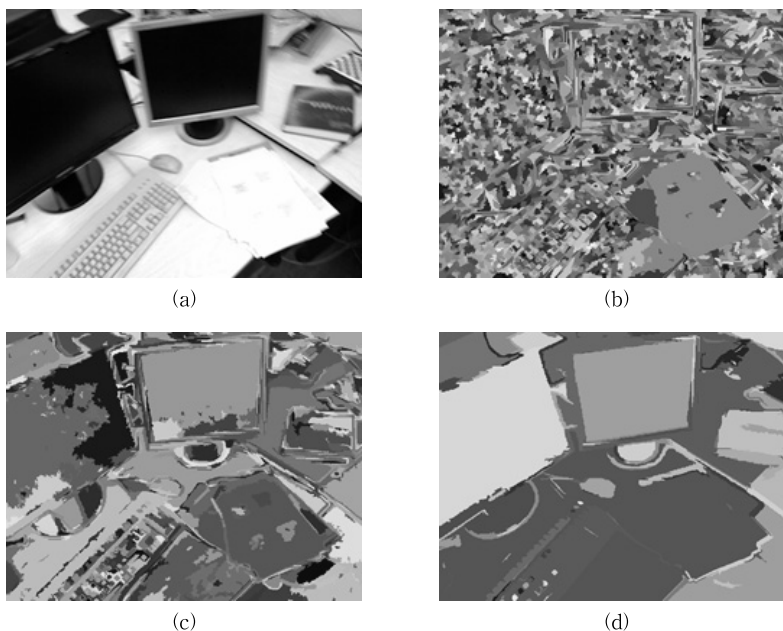


Fig. 8. Color segmented images by different threshold (a) Original image (b) threshold is 0 (c) threshold is 50 (d) threshold is 100

Table 1. Rotation error, translation error, and computational time according to different threshold values and number of samples

data set	sample	Graph-based segmentation threshold								
		0			50			100		
		Rot. (deg)	Transl. (cm)	Time (sec)	Rot. (deg)	Transl. (cm)	Time (sec)	Rot. (deg)	Transl. (cm)	Time (sec)
FR1 desk	100	0.52	0.65	7.69	0.66	1.16	3.65	0.82	1.66	3.04
	500	0.53	0.66	9.28	0.55	0.76	4.60	0.59	0.89	3.96
	1000	0.53	0.65	9.41	0.54	0.70	3.89	0.56	0.78	4.39
	1500	0.53	0.65	9.39	0.54	0.68	4.84	0.55	0.74	4.70
FR1 360	100	0.52	1.55	12.58	0.71	2.19	2.53	0.88	3.07	2.15
	500	0.49	1.73	19.30	0.57	1.50	3.65	0.63	1.87	2.81
	1000	0.49	1.77	19.24	0.54	1.40	4.46	0.61	1.63	3.26
	1500	0.50	1.75	19.14	0.53	1.36	5.07	0.60	1.55	3.66
FR1 floor	100	0.22	0.24	9.06	0.35	0.68	3.59	0.52	1.14	2.43
	500	0.21	0.24	11.90	0.24	0.36	5.23	0.32	0.60	3.38
	1000	0.21	0.24	12.03	0.23	0.30	6.14	0.28	0.46	3.83
	1500	0.21	0.24	12.10	0.22	0.28	6.59	0.26	0.41	4.07

변환벡터를 t_g , 각 알고리즘으로 추정된 자세의 이동변환벡터는 t_e 이다.

4.2. 기존 정합 알고리즘과의 성능 비교

Fig. 9와 Fig. 10은 월드좌표계에서 제안된 방법과 기존의 방법을 비교한 것이다. 비교는 Freiburg 데이터에서 제공되는 groundtruth를 이용하였고 월드좌표계에서 groundtruth와 각 방법으로 추정된 자세의 오차를 계산하였다. 이때 회전 오차는 Equation (7)를 이용하여 계산하고, 이동 오차는 Equation (8)을 이용하여 계산한다.

FR1 desk 데이터의 결과인 Fig. 9를 보면 0~50, 150~250, 400~450 프레임에서 기존의 알고리즘은 오차가 많아지는 반면 본 논문에서 제안한 방법은 비교적 오차가 적음을 볼 수 있다. Table 2에서 볼 수 있듯이 균등 샘플링 기반의 제안된 방법은 회전오차가 0.53°, 이동오차가 0.59cm, 프레임당 24.66초의 계산 속도를 보이며 기존의 알고리즘보다 성능이 개선되었음을 보인다. 제안된 방법을 색상 분할 기반의 샘플링을 했을 경우 회전오차가 0.54°, 이동오차가 0.71cm로 균등 샘플링으로 했을 경우와 큰 성능 차이가 보이지 않는다. 하지만 계산 속도가 3.89초로 약 6.33배 정도 향상된 것을 확인 할 수 있다. 비교적 색상이 고르게 분포되어 있는 환경에서는 색상 분할 기반 샘플링을 이용한 제안된 방법이 성능뿐만 아니라 계산 속도에서 많이 개선되는 모습을 보인다.

FR1 360 데이터 결과인 Fig. 10에서도 기존의 알고리즘들은 0~150, 300~400프레임에서 기존의 알고리즘이 오차가 많이 누적되는 반면에 제안된 알고리즘들은 전반적으로 안정적인 성능을 보인다. Table 3에서도 볼 수 있듯이 각 프레임별 평균 오차도 균등 샘플링을 이용한 제안된 방법은 회전오차가 0.51°, 이동오차가 2.07cm, 계산속도가 24.29초로 기존의 알고리즘보다 개선된 모습을 보인다. 이 실험에서도

색상 분할 기반 샘플링을 이용한 방법이 균등 샘플링 방법과 유사한 이동, 회전오차를 가지는 반면에 계산 속도에서 4.70초로 약 5.21배정도 개선되었음을 알 수 있다. 이 FR1 360 데이터는 프레임간의 변화가 크고 색상 영상에 모션 블러(motion blur)가 심하지만 색상의 분포가 비교적 고르기 때문에 제안된 방법의 성능이 개선된 모습을 보인다.

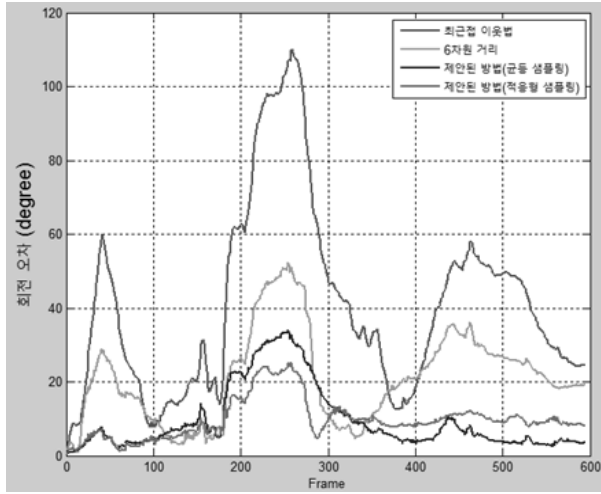
FR1_floor 데이터의 결과인 Table 4에서 보면 각 프레임별 평균 오차가 균등 샘플링을 이용한 제안된 방법은 회전

Table 2. Average error per fame in FR1 desk

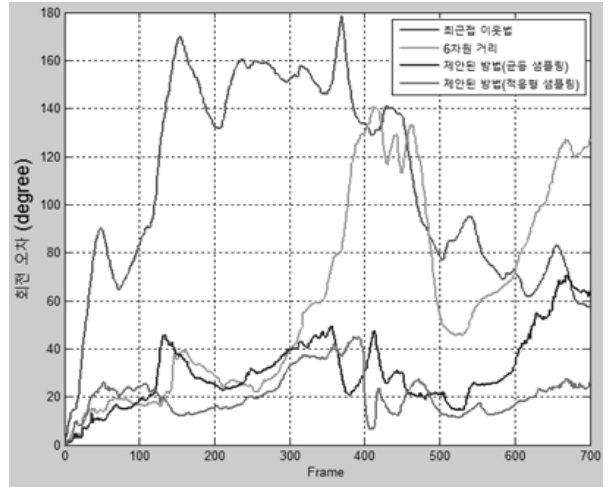
	Rotation (deg)	Translation (cm)	Time (sec/frame)	Average number of sample
Nearest neighbor	1.12	1.45	1.74	229373.18
Standard 6D	0.64	0.68	5.56	229373.18
Proposed (uniform)	0.53	0.59	24.66	229373.18
Proposed (segmentation)	0.54	0.71	3.89	172836.44

Table 3. Average error per fame in FR1 360

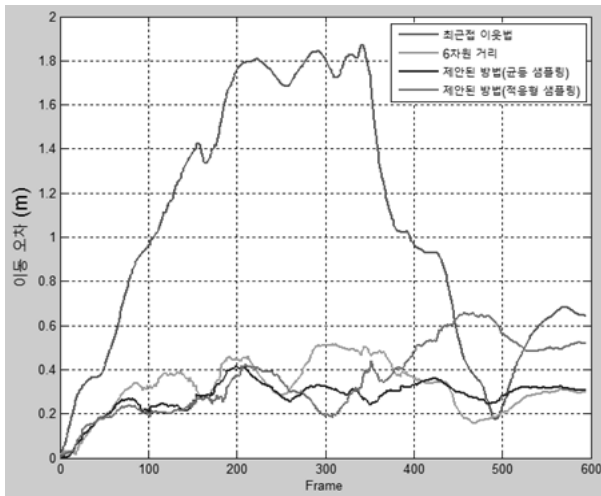
	Rotation (deg)	Translation (cm)	Time (sec/frame)	Average number of sample
Nearest neighbor	1.74	1.37	4.56	243563.34
Standard 6D	0.63	2.32	11.57	243563.34
Proposed (uniform)	0.51	2.07	24.29	243563.34
Proposed (segmentation)	0.58	1.52	4.70	140177.19



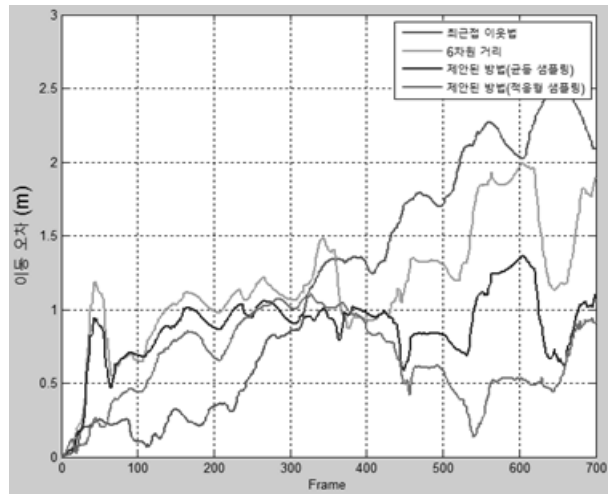
(a)



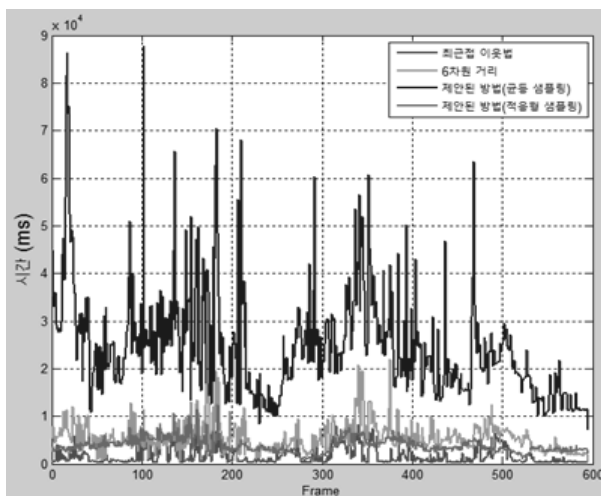
(a)



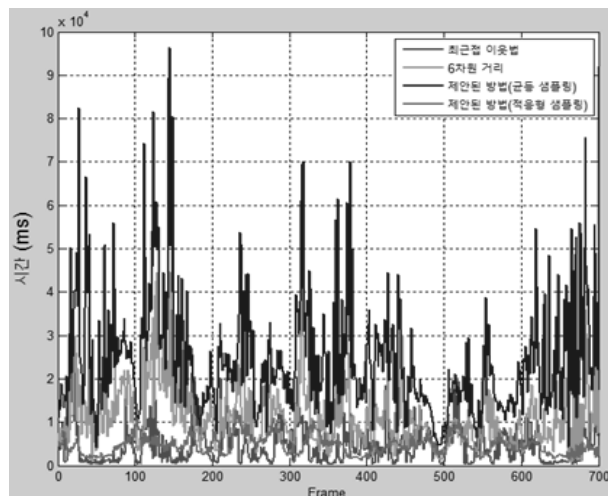
(b)



(b)



(c)



(c)

Fig. 9. Registration error comparison with ground truth of FRI desk dataset (a) Rotation error (b) Translation error (c) Computational time

Fig. 10. Registration error comparison with ground truth of FRI 360 dataset (a) Rotation error (b) Translation error (c) Computational time

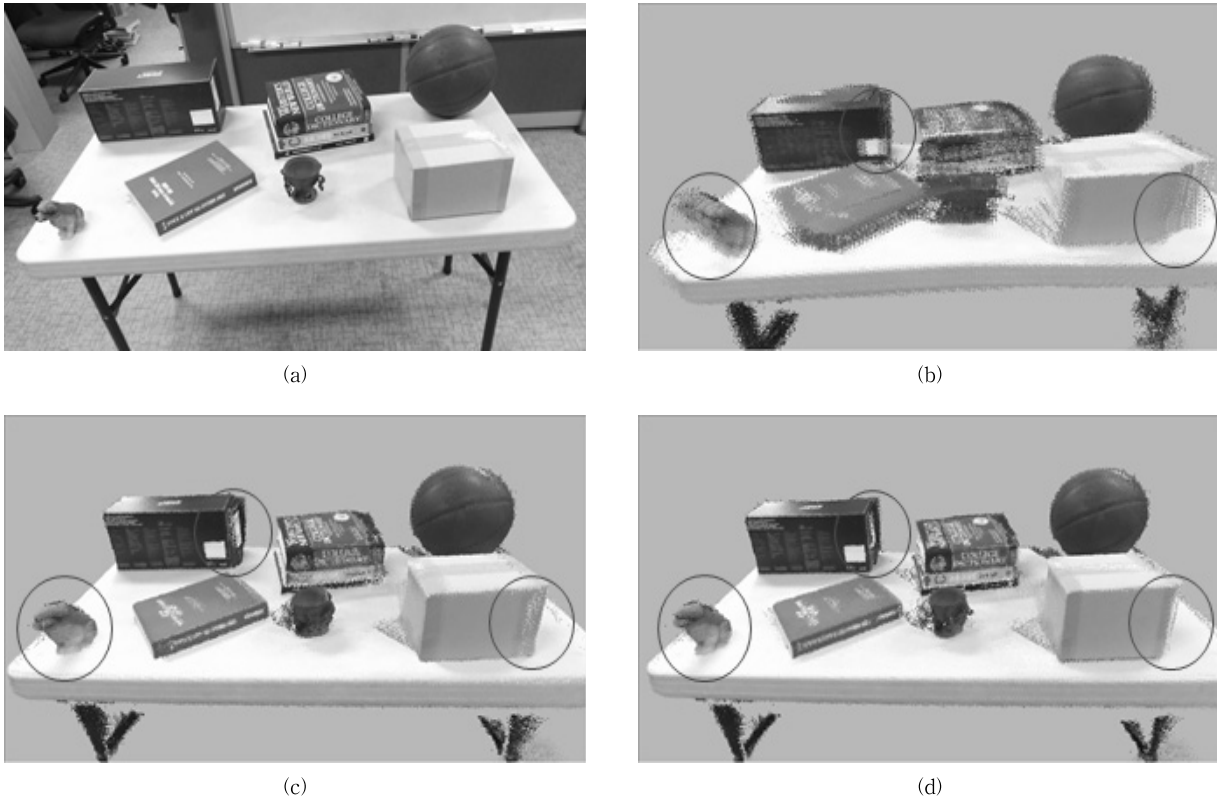


Fig. 11. Registration comparison of different algorithms (a) Target scene (b) Closest neighbor (c) Standard 6D (d) Proposed method

Table 4. Average error per frame in FR1 floor dataset

	Rotation (deg)	Translation (cm)	Time (sec/frame)	Average number of sample
Nearest neighbor	0.59	0.95	0.43	253725.82
Standard 6D	0.29	0.50	5.33	253725.82
Proposed (uniform)	0.24	0.36	17.66	253725.82
Proposed (segmentation)	0.25	0.45	6.15	175412.14

오차가 0.24°, 이동오차가 0.36cm, 계산속도가 17.66초로 기존의 알고리즘보다 개선된 모습을 보인다. 이 실험 결과에서도 확인할 수 있듯이 균등 샘플링 방법과 색상 분할 기반 샘플링의 성능은 거의 유사하지만 계산 속도에서 6.15초로 약 2.87배정도 개선되었음을 보여준다. 이 실험 결과에서 알 수 있듯이 3차원적 특징이 거의 없고 색상이 단조로우에도 불구하고 본 논문에서 제안된 방법이 안정적인 성능을 보여준다.

위 실험을 통해 본 논문에서 제안한 방법이 기존의 알고리즘들에 비해 프레임간의 변화가 큰 데이터에서도 성능이 안정적이고 3차원적 특징 및 색상이 단조로운 데이터에서도

강인함을 실험을 통해 알 수 있었다. 또한 색상 분할 기반의 샘플링으로 인해 기존의 알고리즘들에 비해 계산속도가 유사하거나 개선된 모습을 확인할 수 있었다.

4.3 Kinect v2 데이터 실험

실제 정합된 결과를 최근에 마이크로소프트사에서 출시된 Kinect v2 를 이용하였다. RGB 영상은 1920x1080 해상도로 깊이 영상은 512x424 해상도로 획득이 가능하다. 깊이 카메라의 수평 시야각은 70°, 수직 시야각은 60°이다.

Fig. 11a는 실제 실험에 사용된 3차원 모델을 촬영한 RGB 영상이다. 다양한 물체가 올려져있는 탁자 위를 이동하며 촬영한 영상으로써 총 300프레임에 해당된다. 이 데이터를 이용하여 최근접 이웃법, 6차원 거리, 색상 분할 기반 적응형 샘플링을 이용한 제안된 방법 등 총 3가지의 방법을 이용하여 3차원 정합을 하였다. Fig. 11b는 최근접 이웃법을 이용한 정합 결과이다. 이 영상에서도 알 수 있듯이 많은 누적 오차가 발생하는 모습을 볼 수 있다. Fig. 11c는 6차원 거리를 이용한 것으로 최근접 이웃법에 비해 누적오차가 많이 없어졌음을 볼 수 있다. Fig. 11d는 본 논문에서 제안된 방법에 대한 정합 결과 이다. 붉은 동그라미로 표시된 부분을 보면 6차원 거리를 이용한 정합에 비해 제안된 방법이 누적오차가 적음을 알 수 있다.

5. 결 론

본 논문에서는 6차원 거리와 색상분포 유사도를 이용한 새로운 대응점 탐색방법을 통해 기존의 3차원 정합 기술의 성능을 개선하는 방법을 제안했다. 먼저 6차원 거리를 이용하여 대응점 후보를 탐색하고 색상분포 유사도를 비교를 통해 최종적인 대응점을 결정한다. 이때 색상분포 유사도는 특이값 분해를 이용하여 벡터연산으로 계산한다. 하지만 본 논문이 제안하는 방법은 계산에 복잡하기 때문에 기존의 방법들에 비해 많은 시간이 소요했다. 이를 극복하는 방안으로 색상 분할 기반 적응형 샘플링을 이용하여 제안된 방법의 속도를 향상시켰다. 색상 영상을 그래프 기반 분할 방법으로 영역을 분할하고 각 영역별로 무작위로 샘플링을 하는데 이때 선택할 샘플 수와 영역의 크기를 결정하는 임계값은 실험을 통해 알아낸다. 본 논문에서 제안한 방법의 성능 확인하기 위해 검증된 데이터를 사용하여 실험하였다. 실험을 통해 본 논문에서 제안된 방법이 기존의 3차원 정합들보다 성능이 향상되었음을 보였다. 특히 프레임간의 변화 큰 데이터라도 색상의 분포가 다양하다면 안정적인 성능을 보장 받을 수 있다는 것을 확인 할 수 있었다. 반대로 색상 분할 기반 적응형 샘플링은 촬영된 RGB 영상의 색상이 다양하지 않게 되면 안정적인 성능을 보장 받을 수 없다. 향후에는 이러한 문제점을 보완하여 제안된 방법을 보다 개선하기 위한 연구를 진행할 계획이다.

References

[1] P. Henry, M. Krainin, E. Herbst, X. Ren, and D. Fox, "RGB-D mapping: Using Kinect-style depth cameras for dense 3D modeling of indoor environments," *The International Journal of Robotics Research*, Vol.31, No.5, pp.647-663, 2012.

[2] C. Choi, A. J. B. Trevor, and H. I. Christensen, "RGB-D edge detection and edge-based registration," in *Proceedings of the IEEE International Conference on Intelligent Robots and Systems*, 2013.

[3] T. Whelan, H. Johannsson, M. Kaess, J. J. Leonard, and J. McDonald, "Robust real-time visual odometry for dense RGB-D mapping," in *Proceedings of the IEEE International Conference on Robotics and Automation*, 2013.

[4] P. J. Besl and N. D. MaKay, "Method for registration of 3-D shapes," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol.14, No.2, 1992.

[5] M. Rouhani and A. D. Sappa, "Correspondence free registration through a point-to-model distance minimization," in *Proceedings of the IEEE international conference on Computer Vision*, 2011.

[6] A. Censi, "An ICP variant using a point-to-line metric," in *Proceedings of the IEEE International Conference on Robotics and Automation*, 2008.

[7] Y. Chen and G. Mdeioni, "Object modeling by registration of multiple range images," in *Proceedings of the IEEE International Conference on Robotics and Automation*, 1991.

[8] A. Segal, D. Haehnel, and S. Thrun, "Generalized-ICP," *Robotics: Science and Systems*, Vol.2, No.4, 2009.

[9] A. E. Johnson and S. B. Kang, "Registration and integration of textured 3D data," in *Proceedings of the IEEE International Conference on 3-D Digital Imaging and Modeling*, 1997.

[10] H. Men, B. Gebre, and K. Pochiraju, "Color point cloud registration with 4D ICP algorithm," in *Proceedings of the IEEE International Conference on Robotics and Automation*, 2011.

[11] M. Korn, M. Holzkothen, and J. Pauli, "Color Supported Generalized-ICP," in *Proceedings of the International Conference on Computer Vision Theory and Applications*, 2014.

[12] J. H. Joung, K. H. An, J. W. Kang, M. J. Chung, and W. Yu, "3D environment reconstruction using modified color ICP algorithm by fusion of a camera and a 3D laser range finder," in *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2009.

[13] L. Douadi, M.-J. Aldon, and A. Crosnier, "Pair-wise Registration of 3D/Color Data Sets with ICP," in *Proceedings of the IEEE International Conference on Intelligent Robots and Systems*, 2006.

[14] F. T. Romero and M. Devy, "Registration by using a pseudo color attribute," in *Proceedings of the IEEE International Conference on Pattern Recognition*, 2008.

[15] S. Rusinkiewicz and M. Levoy, "Efficient variants of the ICP algorithm," in *Proceedings of the IEEE International Conference on 3-D Digital Imaging and Modeling*, 2001.

[16] P. F. Felzenszwalb and D. P. Huttenlocher, "Efficient Graph-Based Image Segmentation," *International Journal of Computer Vision*, Vol.59, No.2, pp.167-181, 2004.

[17] J. Sturm, N. Engelhard, F. Endres, W. Burgard, and D. Cremers, "A benchmark for the evaluation of RGB-D SLAM systems," in *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2012.

[18] P. F. Felzenszwalb and D. P. Huttenlocher, "Efficient Graph-Based Image Segmentation" [Online], Available: <http://cs.brown.edu/~pff/segment/>. [Accessed 2 April 2015].



김응수

e-mail : jsm80607@naver.com
 2013년 대구대학교 컴퓨터IT공학부(학사)
 2015년 경북대학교 컴퓨터학부(석사)
 2015년~현 재 경북대학교 컴퓨터학부
 박사과정
 관심분야 : 3차원 정합 및 복원, SLAM



최성인

e-mail : ellim5th@naver.com

2007년 금오공과대학교 컴퓨터학부(학사)

2009년 경북대학교 컴퓨터공학과(석사)

2015년 경북대학교 컴퓨터학부(박사)

2015년~현 재 삼성중공업 중앙연구소
책임연구원

관심분야: 3차원 정합 및 복원, 로봇비전



박순웅

e-mail : sypark@knu.ac.kr

1991년 경북대학교 전자공학과(학사)

1993년 경북대학교 전자공학과(석사)

2003년 미국 뉴욕 주립대학교 스토니브룩
(박사)

2005년~현 재 경북대학교 컴퓨터학부 교수

관심분야: 3차원 스캐닝, 3차원 정합, 로봇비전