

포인트 클라우드 정합 성능 향상을 위한 리샘플링 방법

김종욱, 박종일[†]

한양대학교

{glowingbell, jipark}@hanyang.ac.kr

Resampling Method to Improve Performance of Point Cloud Registration

Jongwook Kim Jong-Il Park[†]

Hanyang University

요 약

본 논문에서는 포인트 클라우드 정합 성능 향상을 위해 기하적 복잡도가 낮은 정점들의 영향을 최소화하는 포인트 클라우드 리샘플링 방법을 제안한다. 3 차원 특징 기술자(3D feature descriptor)를 기반으로 하는 포인트 클라우드 정합은 정점 법선 벡터의 변화량을 특징으로 사용한다. 따라서 강건한 특징은 대부분 정점 법선 벡터의 변화량이 큰 영역에서 추출된다. 반면에 정점 법선 벡터의 변화량이 거의 없는 평면 영역은 정합 수행 시에 이상점(outlier)으로 작용할 수 있으므로 해당 정점들이 정합 과정에 미치는 영향을 최소화해야 한다. 제안하는 방법은 모델 포인트 클라우드의 기하적 복잡도를 고려한 리샘플링을 통해 전체 정점의 수 대비 복잡도가 낮은 정점들의 비율을 낮추어 이상점이 정합 과정에 미치는 영향을 최소화하고 정합 성능을 향상시켰다.

1. 서론

최근 RGB-D 카메라와 LIDAR 등 3 차원 정보 획득 장비들의 발전으로 이를 활용한 컴퓨터 비전 기반 객체 인식, 자세 추정 등의 연구가 활발히 진행되고 있다. 객체 인식, 자세 추정 연구는 다양한 산업 분야에서 적용이 이루어지고 있는데 특히, 3 차원 그래픽을 실세계의 객체 위에 증강하는 증강현실 분야와 로봇으로 물건을 집어내는 공정 자동화 분야에서 핵심 기술로 사용되고 있다.

객체를 인식하고 자세를 추정하기 위해서는 사전에 알고 있는 3 차원 메쉬 모델로부터 획득한 모델 포인트 클라우드를 센서에서 획득한 포인트 클라우드에 정합해야 한다. 포인트 클라우드 정합에는 3 차원 특징 기술자(3D feature descriptor)가 사용되는데, 3 차원 특징 기술자는 일정 탐색 영역의 정점 법선 벡터의 변화량을 특징으로 사용한다[1][2]. 따라서 정점 법선

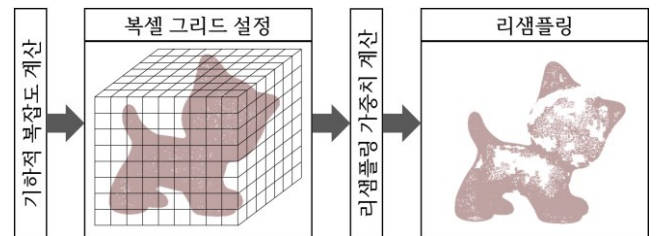


그림 1. 제안하는 방법의 흐름도

벡터의 변화량이 큰 영역일수록 강건한 특징이 추출된다. 반면에 정점 법선 벡터의 변화량이 거의 없는 평면 영역은 특징이 모호하기 때문에 정합 수행 시에 이상점(outlier)으로 작용할 수 있다. 즉, 정합 과정에서 이상점으로 작용할 수 있는 정점들을 포함한다면 불필요한 연산이 수반되어 성능의 저하를 초래한다.

이에 본 논문에서는 정점 법선 벡터의 변화량을 기하적 복잡도로 정의하고, 기하적 복잡도가 낮은 영역에서 발생할 수 있는 이상점을 최소화하는 방법을 제안한다. 제안하는 방법으로

[†]교신저자

전체 정점의 수 대비 복잡도가 낮은 정점들의 비율을 낮추어 이상점이 정합 과정에 미치는 영향을 최소화하였고, 실험을 통해 정합 성능이 향상됨을 보였다.

2. 기하적 복잡도를 고려한 리샘플링 방법

2 장에서는 모델의 기하적 복잡도를 고려한 포인트 클라우드 리샘플링 방법에 대해서 설명한다. 본 논문에서 제안하는 리샘플링 방법은 다음과 같다. 먼저 전체 모델 포인트 클라우드의 정점에 대해서 기하적 복잡도를 계산하고, 모델 포인트 클라우드를 일정 크기의 복셀 그리드로 나눈다. 그 다음 각각의 복셀 그리드 영역에 속한 정점들의 기하적 복잡도를 가중치로하여 리샘플링을 수행한다.

2.1 기하적 복잡도 계산

기하적 복잡도를 계산하기 위해서는 우선 3 차원 메쉬 모델에서 포인트 클라우드를 샘플링하고 정점 법선 벡터를 계산해야 한다. 모델 포인트 클라우드 샘플링은 균등 분포(uniform distribution)에 따라 메쉬 모델을 이루는 폴리곤 영역을 무작위로 선택하고, 선택된 폴리곤 영역에서 무작위로 점을 샘플링하는 Osada 등이 제안한 방법을 적용하였다[3]. 정점 법선 벡터는 무작위로 선택된 폴리곤의 세 정점을 사용하여 계산하였다. 기하적 복잡도는 다음 식에 의해 정의된다.

$$\kappa_i = \frac{1}{N} \sum_{j=0}^N \cos^{-1} \left(\frac{v_i \cdot v_j}{\|v_i\| \|v_j\|} \right) \quad (1)$$

모델 포인트 클라우드 상에 존재하는 임의의 한 정점 p_i 의 법선 벡터 v_i 와 p_i 의 일정 범위 내에 존재하는 이웃한 정점 p_j 의 법선 벡터를 v_j 라 할 때, 정점 p_i 의 기하적 복잡도는 κ_i 이다. N 은 정점 p_i 의 일정 범위 내에 존재하는 이웃한 정점 p_j 의 수를 나타낸다.

2.2 리샘플링 가중치 계산

모델 포인트 클라우드를 그림 1 과 같이 일정 크기의 복셀 그리드로 나누고, 각 복셀 그리드 영역에 속한 정점들의 기하적 복잡도에 따라 다음 식과 같이 해당 그리드의 리샘플링 가중치를 결정한다.

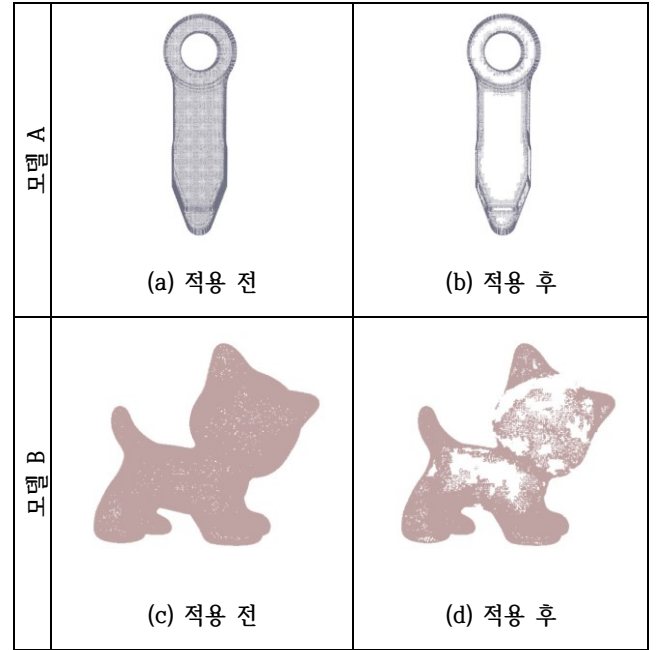


그림 2. 리샘플링 적용 전과 후 결과 비교

$$w_k = \frac{1}{M} \sum_{i=0}^M \kappa_i \quad (2)$$

κ_i 는 식 1 에서 계산한 기하적 복잡도를 의미하고, w_k 는 해당 복셀 그리드의 가중치를 나타낸다. M 은 해당 그리드 영역에 속한 정점의 수를 나타낸다. 본 논문에서는 모델의 전체 가중치 분포에서 하위 40%의 가중치를 갖는 정점을 제외하고 리샘플링 하였다.

3. 실험 결과

본 논문에서는 Intel i7-4790 CPU 환경에서 정합 속도를 측정하였고, Microsoft Azure Kinect 카메라를 사용하여 실험에 필요한 포인트 클라우드를 획득하였다[4]. 그림 2 는 실험에 사용한 모델들의 리샘플링 적용 전과 후의 결과이다. 전체적으로 낮은 기하적 복잡도를 갖는 모델 A 와 높은 기하적 복잡도를 갖는 모델 B 를 실험에 사용하였다[5].

그림 3 은 제안하는 리샘플링 방법을 적용한 경우와 적용하지 않은 경우의 정합 결과 영상이다. 제안하는 리샘플링 방법을 적용하지 않은 정합 결과 영상에서 대부분의 대응점이 기하적 복잡도가 높은 영역에서 형성되는 것을 확인할 수 있다. 특히, 전체적으로 낮은 기하적 복잡도를 갖는 모델 A 의 경우, 평면 영역에서 대응점이 거의 형성되지 않은 것을 알 수 있다.

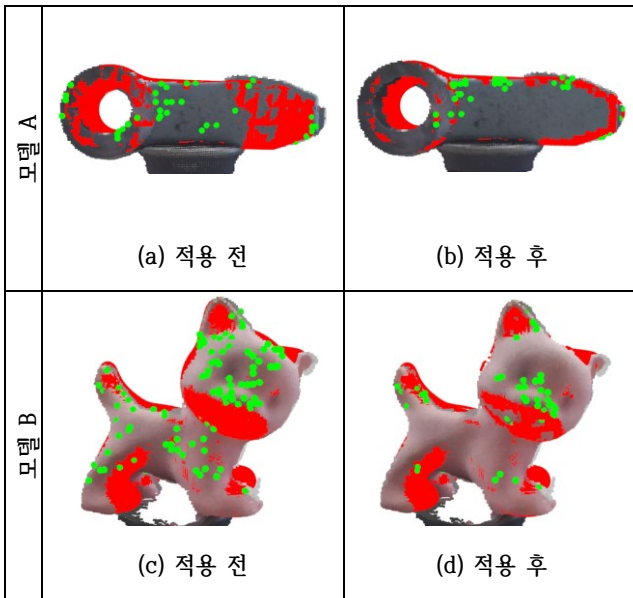


그림 3. 리샘플링 적용 전과 후 정합 결과 비교 (모델 포인트 클라우드: 적색 포인트 클라우드, 매칭된 대응점: 녹색 점)

표 1. 리샘플링 적용 전과 후의 정합 성능 비교

모델	방법	정점의 수	정합 오차	정합 시간
모델 A	적용 전	32,678	1.541 mm	310 ms
	적용 후	20,617	1.359 mm	145 ms
모델 B	적용 전	43,169	1.801 mm	336 ms
	적용 후	26,073	1.789 mm	246 ms

표 1은 제안하는 리샘플링 방법을 적용한 경우와 적용하지 않은 경우의 정점의 수와 정합 성능을 비교한 결과이다. 일반적으로 포인트 클라우드 정합에 사용되는 정점의 수가 감소함에 따라 정합 속도는 향상되지만, 정합 오차는 증가하는 경향이 있다. 하지만 실험 결과 제안하는 방법을 정합에 적용했을 때, 정점의 수가 감소했음에도 불구하고 정합 오차에 영향을 미치지 않고 정합 속도가 효과적으로 향상됨을 확인할 수 있다.

4. 결론

본 논문에서는 포인트 클라우드 정합 성능 향상을 위한 포인트 클라우드 리샘플링 방법을 제안하였다. 제안하는 방법은 모델의 기하적 복잡도를 고려한 포인트 클라우드 리샘플링을 통해 기하적 복잡도가 낮은 영역의 비율을 낮추어 이상점이 정합 과정에 미치는 영향을 최소화하였다. 또한 제안하는 리샘플링

방법을 적용한 포인트 클라우드 정합 실험을 통해 정합 성능이 향상됨을 보였다.

감사의 글

이 논문은 2020년도 정부(과학기술정보통신부)의 재원으로 정보통신기술진흥센터의 지원을 받아 수행된 연구임 (No.2017-0-01849, 실내의 임의공간 실시간 영상 합성을 위한 핵심 원천기술 및 개발툴킷 개발)

참고문헌

- [1] Tombari Federico, Samuele Salti, and Luigi Di Stefano. "Unique signatures of histograms for local surface description." in *European Conference on Computer Vision*. Springer, Berlin, Heidelberg, 2010, pp. 356-569.
- [2] Yulan Guo, Mohammed Bennamoun, Ferdous Sohel, Min Lu, Jianwei Wan, and Ngai Ming Kwok. "A comprehensive performance evaluation of 3D local feature descriptors." *International Journal of Computer Vision*, 2016, 116.1: 66-89
- [3] Robert Osada, Thomas Allen Funkhouser, Bernard Marie Chazelle, and David P Dobkin. "Shape distributions." *ACM Transactions on Graphics (TOG)*, 21.4: 807-832.
- [4] Azure Kinect DK, Microsoft. Accessed: Oct. 11, 2020. [Online]. Available: <https://docs.microsoft.com/ko-kr/azure/Kinect-dk/>
- [5] Stefan Hinterstoisser, Vincent Lepetit, Slobodan Ilic, Stefan Holzer, Gary Bradski, Kurt Konolige, and Nassir Navab. "Model based training, detection and pose estimation of texture-less 3D objects in heavily cluttered scenes." in *Asian Conference on Computer Vision (ACCV)*, 2012, pp. 548-562.