

연구논문

# LIDAR 포인트 cloud로부터 분리된 포인트 군집간 인접관계 인식과 응용에 관한 연구

## A Study on Detecting Neighboring Relation Among Point Segments of LIDAR Point Cloud and its Application

한수희\* · 이정호\*\* · 유기윤\*\*\* · 김용일\*\*\*\*

Han, Soo Hee · Lee, Jeong Ho · Yu, Ki Yun · Kim, Yong Il

### 要 旨

본 연구에서는 선행 연구에서 제시한 스캔라인을 이용한 LIDAR 포인트 cloud의 분리과정 중 분리된 포인트 군집간 인접 관계를 인식할 수 있는 기능을 추가하여 분리된 건물요소를 인식하고 재결합하였다. 아울러 군집간 인접 특성을 활용하여 지면 포인트 군집을 인식하기 위한 방법론을 제시하였다. 실험 결과 포인트 cloud 분리 과정에 군집간 인접 관계 인식 기능을 추가하더라도 처리 성능이 저하되지 않았으며 후처리를 통하여 건물 요소를 결합하여 온전한 형태의 건물 포인트 군집을 형성함과 더불어 지면 포인트 군집도 인식할 수 있음을 확인하였다.

**핵심용어** : LIDAR, 포인트 cloud, 분리, 인접 관계

### Abstract

In this study, we recognized and joined parts of buildings separated during LIDAR point segmentation utilizing scan line characteristics, with an additional function to recognize neighboring relation among point segments. And we applied the relation to suggest a method to recognize earth point segment. From the test, we could confirm that it does not drop down the efficiency of point segmentation to be added with the function of recognizing neighboring relation and it is possible to combine point segments to form a complete shaped building and to recognize earth point segment.

**Keywords** : LIDAR, Point cloud, Segmentation, Neighboring relation

## 1. 서 론

광역적인 3차원 공간정보 구축을 위하여 LIDAR 포인트 cloud를 효율적으로 분리하기 위한 연구를 본 연구에 앞서 수행하였다. 선행 연구에서 제시한 LIDAR 포인트 cloud 분리 방식(한수희 등, 2005, 2006)은, region growing과 무감독 분류와 유사한 방식으로 원시 LiDAR 포인트 자료를 분리하였으며 처리 효율성을 증진시키기 위하여 항공 레이저 측량 장비의 스캔라인 특성을 활용하였다. 결과적으로 대량의 포인트 자료를 매우 효과적으로 처리할 수 있었으며 분리 정확도도 합리적인 수준으로 나타났다. 여기서, 일정 거리 이상 떨어져 있거나 고도 차이가

일계치 이상인 개체는 서로 분리되어 각각의 포인트 군집을 이루게 된다. 따라서 고도가 다른 건물요소들로 구성된 건물의 경우 여러 개의 포인트 군집으로 나뉘어 분리된다.

본 연구에서는 이와 같이 분리된 건물요소를 인식하고 재결합하기 위하여 선행 연구의 포인트 분리 과정 중에 군집간 인접 관계를 파악할 수 있는 기능을 추가하였다. 인접 관계에 있는 군집들은 후처리 과정을 통하여 인접 관계가 재정립되어 최종적으로 재결합된다. 아울러 대부분의 지상물은 지면과 인접관계에 있다는 특성을 이용하여, 지면 포인트 군집을 인식하는 방법론을 제시하였다. 본 논문은 선행 연구의 주요 알고리즘의 간략한 소개와

2006년 11월 27일 접수, 2007년 1월 19일 채택

\* 서울대학교 공과대학 지구환경시스템공학부 박사과정 (scivile2@snu.ac.kr)  
\*\* 서울대학교 공과대학 지구환경시스템공학부 박사과정 (ilome79@snu.ac.kr)  
\*\*\* 정희원, 서울대학교 공과대학 지구환경시스템공학부 조교수 (kiyun@snu.ac.kr)  
\*\*\*\* 정희원, 서울대학교 공과대학 지구환경시스템공학부 부교수 (yik@snu.ac.kr)

군집간 인접관계를 파악하기 위해 보완된 내용, 후처리 과정과 적용 및 평가의 순서로 구성되어 있다.

## 2. 본 론

### 2.1 선행 연구의 포인트 cloud 분리

선행 연구의 포인트 cloud의 분리 방식은 다음과 같다 (그림 1). 스캔라인을 따라 입력되는 신규 포인트는 기본적으로 새로이 생성될 포인트 군집의 seed 포인트가 된다 (step6) 만약 기 생성된 군집에 신규 포인트와의 고도 차이와 거리 차이(step 3)가 주어진 임계치 이하인 인접 포인트가 존재할 경우 신규 포인트는 새로운 군집을 형성하지 않고 경우 해당 군집으로 분류된다(step 4, 5). 한편, 신규 포인트가 두 개 이상의 군집으로 분류 가능할 경우 해당 군집들은 하나의 개체로 인식되어 통합된다(step 6).

```

step 1 : initialize n=1
step 2 : initialize  $L_{merge}$  as empty and  $nc=i=0$ 
step 3 : calculate  $d_{ni}=d(p_{new}, p_{mi})$  and  $h_{ni}=h(p_{new}, p_{mi})$ 
step 4 : if ( $d_{ni} \leq t_{dist} \& h_{ni} \leq t_{height}$ ) then  $nc=nc+1$ 
           go to step 3 for  $\forall p_{mi} \in G_i$ 
step 5 : if ( $nc=1$ ) then classify  $p_{new}$ 
           into  $G_i$  and put  $i$  into  $L_{merge}$ 
           else if ( $nc>1$ ) then put  $i$  into  $L_{merge}$ 
           set  $nc=0$  and go to step 3 for all
            $G_i (i = 1, \dots, n)$ 
step 6 : if ( $n(L_{merge})=0$ ) then
            $n=n+1$ , create  $G_n$  and classify  $p_{new}$ 
           into  $G_n$ 
           else if ( $n(L_{merge})>1$ ) then merge
           groups listed in  $L_{merge}$ 
           repeat from step 2 to step 6 for  $\forall p_{new} \in \{point\ cloud\}$ 

```

그림 1. 포인트 cloud 분리 알고리즘

여기서, n: 총 군집 수,  $p_{new}$ : 신규 포인트,  $p_{mi}$ : i군집의 m번째 포인트,  $G_i$ : i번째 군집,  $d(p_1, p_2)$ 와  $h(p_1, p_2)$ : 포인트  $p_1, p_2$ 의 거리와 높이차,  $t_{dist}$ 와  $t_{height}$ : 거리 및 높이 차이 임계치,  $L_{merge}$ : 통합될 군집 목록

### 2.2 보완된 포인트 cloud 분리 방식

군집간 인접 관계를 파악하기 위하여 다음과 같은 조건과 과정을 추가하였다. 신규 포인트가 어떤 군집에서 한 개 이상의 포인트와 인접하나 고도 차이가 임계치 이하인 포인트가 존재하지 않는 경우, 해당 군집은 신규 포인트가 최종적으로 분류되는 군집이나 seed 포인트가 되는 군집과 인접관계를 갖고 있는 것으로 인식된다. 이와 같이 인접 신규 포인트에 의하여 인접 관계에 있는 것으로

인식된 군집은 인접 군집 목록에 입력된다. 따라서 (그림 1)의 “step 4”를 (그림 2)와 같이 수정, 보완하였다.

```

step 4
if ( $d_{ni} \leq t_{dist} \& h_{ni} \leq t_{height}$ ) then  $nc=nc+1$ 
else if ( $d_{ni} \leq t_{dist} \& h_{ni} > t_{height}$ ) then put  $i$  into  $N_{temp}$ 
go to step 3 for  $\forall p_{mi} \in G_i$ 

```

그림 2. 수정된 step 4

여기서,  $N_{temp}$ : (임시)인접 군집 목록

다음으로 (그림 3)과 같이 “step 7”을 추가하여, 신규 포인트가 최종적으로 분류된 군집에 인접 군집 목록을 첨부한다. 또한 인접 관계에 있는 군집들의 인접 목록도 갱신하여 인접 군집 간에 인접 정보를 상호 일치시킨다.

```

step 7
copy  $N_{temp}$  to  $N_k (p_{new} \in G_k)$  and put  $k$  into  $\forall N_j (j \in L_k)$ 
empty out  $N_i$ 

```

그림 3. 추가된 step 7

여기서,  $N_k, N_j$ : k군집 과 j군집의 인접 군집 목록

군집간 인접 관계 파악 방식을 도식화 하면 (그림 4)와 같다. (그림 4)에서 스캔라인을 따라 포인트 p\_A가 입력 되면 p\_A는 group A로 분류될 수 있다. 다음으로 입력되는 포인트 p\_B는 group A의 포인트들과 일정 수준 이상의 고도 차이가 나므로 group A로 분류될 수 없다. 따라서 p\_B는 새로운 group B의 seed point가 된다. 그러나

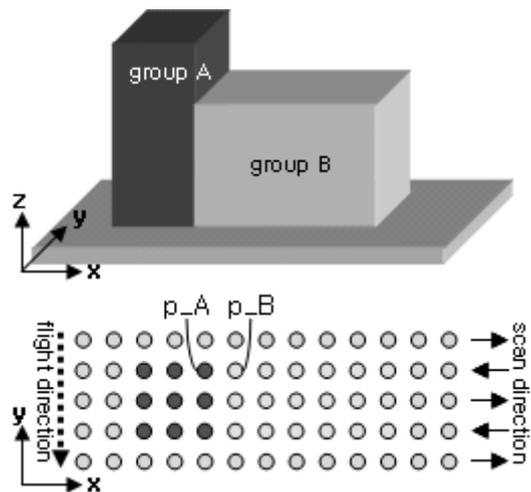


그림 4. 개체간 인접 관계의 파악

p\_B는 p\_A와 인접해 있으므로 group B와 group A는 서로 인접해 있는 것으로 인식되며 group A와 group B에 서로에 대한 인접 정보가 추가된다.

한편, “step 6”에서와 같이 어떤 군집이 다른 군집으로 통합될 경우 인접 정보 역시 통합되고 관련된 군집들의 인접 군집 정보도 갱신한다.

**2.3 군집간 인접 정보를 이용한 후처리**

군집간 인접 정보를 이용한 분리후처리 과정은 다음과 같이 구성된다.

- 지면 포인트 군집 인식
- 군집간 인접 정보 통합 및 재레이블링
- 삭제 개체를 고려한 인접 관계 재정립

**2.3.1 지면 포인트 군집 인식**

기 제시한 포인트 cloud 분리 방식에는 지면을 인식하는 기능은 포함되어 있지 않다. 즉, 지면 군집이 분리되어 있기는 하나 어느 군집이 지면을 나타내는지는 명시되지 않은 상태로 종결되는 것이다. 현재까지 지면 포인트를 인식하기 위한 다수의 필터링 방식이 개발되었으나 본 연구에서는 군집간 인접 정보만으로 간단히 지면 군집을 인식하는 방법론을 제시하였다.

일반적으로 지상의 개체들은 일부 건물요소를 제외하고 대부분 지면과 인접 관계에 놓여 있다. 따라서 대상 지역이 충분히 크고 지면이 축대나 기타 인공구조물로 인하여 작은 부분으로 나뉘어 있지 않는 한 지면은 상대적으로 매우 많은 수의 인접 관계를 갖게 된다. 실제로 300여만 개의 샘플 포인트를 처리하였을 경우 (표 1)과 같은 결과를 나타내었으며 지면 군집의 경우 다른 어느 군집에 비하여 앞도적인 인접 관계를 가지고 있음을 확인할 수 있었다.

**2.3.2 군집간 인접 정보 통합 및 재레이블링**

군집간 인접 정보는 직접적인 인접 관계만을 표현하므로 간접적으로 형성되는 인접 관계를 표현해 주어야 한다. 예를 들어, 군집 A와 B가, B와 C가 인접 관계에 있는

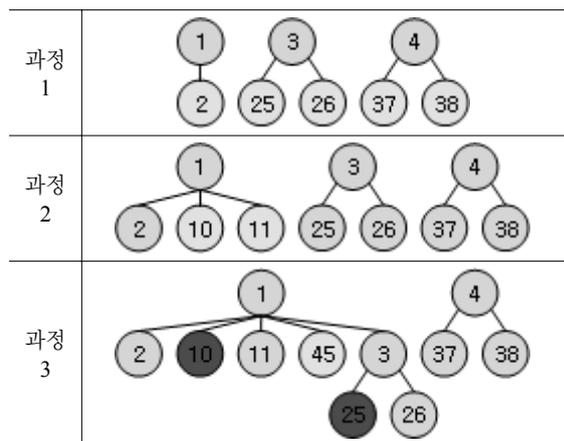
것으로 인식되면 최종적으로 A, B, C가 인접 관계에 있는 것으로 인식시키는 과정이 필요하다. 이를 위하여 다음과 같이 트리 구조를 응용한 처리 방식을 채택하였다.

- 과정 1: 기본적으로 모든 군집은 독립적인 트리의 최상위 노드가 되며 인접 군집은 하위 노드가 됨
- 과정 2: 어떤 군집이 이미 다른 군집의 하위 노드로 판명된 경우 새로이 트리를 형성하지 않으며 해당 군집과 인접한 군집은 최상위 노드의 하위 노드가 됨
- 과정 3: 하위 노드 간에 인접 관계가 있는 것으로 판명될 경우 해당 최상위 노드는 다른 최상위 노드의 하위 노드가 됨

(표 2)와 같은 인접 정보가 주어진 경우를 예로 들면, 각 과정은 (그림 5)와 같이 도식적으로 표현할 수 있다. 이와 같은 과정을 통하여 건물 요소를 인식하면 각 최상위 노드에는 새로운 ID를 부여하고 하위 노드에는 보조 ID를 부여함으로써 재레이블링을 수행한다(그림 6, 7).

**표 2. 군집간 인접 관계**

군집 ID	인접 군집 ID 목록
1	2
2	1, 10, 11
3	25, 26
4	37, 38
:	:
10	25, 45
:	:



**그림 5. 인접 정보의 통합**

**표 1. 지면 군집의 인접 군집 수**

총 군집 수	44,198개
인접 군집 수 평균	약 5개
인접 군집 수 중앙값	3개
인접 군집 수 표준편차	약 125
지면 군집의 인접 군집 수	26,259개

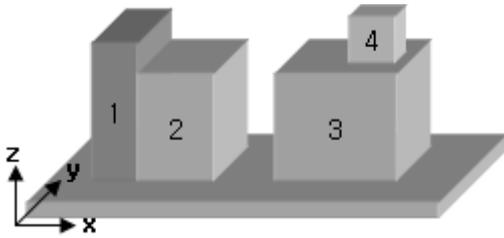


그림 6. 분리된 건물 요소

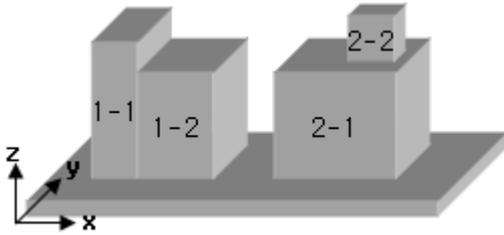


그림 7. 건물 요소의 인식과 재레이블링

2.3.3 삭제 군집을 고려한 인접 관계 재정립

독립된 건물이 인접한 오차 포인트나 수목 등의 불필요 군집에 의해 건물 요소로 인식되는 경우가 발생할 수 있

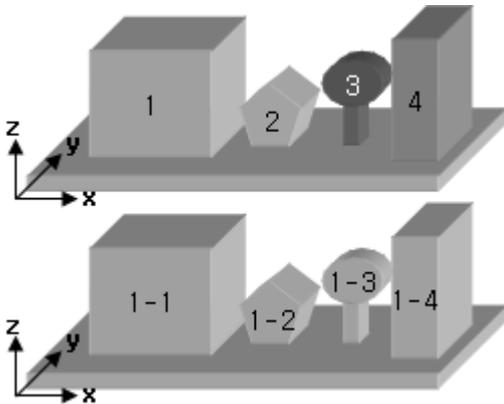


그림 8. 불필요 군집에 의해 독립된 건물이 건물 요소로 인식된 경우



그림 9. 불필요 개체 삭제 및 인접 관계의 재정립

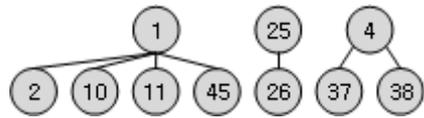


그림 10. 인접 관계의 재정립

다(그림 8). 이와 같은 경우 불필요 군집을 제거한 후에 군집간 인접 관계를 재정립 한다(그림 9).

예를 들어 (그림 5)의 과정 3에서 만약 3번 노드가 불필요 군집으로 판명될 경우, 3번 노드의 하위 노드였던 25번 노드와 26번 노드는 분리되어 독립적인 트리를 구성하게 된다.

오차 포인트나 특정 대상물의 인식은 여러 가지 방식으로의 접근이 가능하다. 그 중, Sithole(2005)은 n개의 인접한 포인트들로부터 평면 방정식을 구하여 표면 거칠기를 산출함으로써 수목을 인식하였고, 이정호(2005)는 스캔라인을 따라 1차원 필터링을 적용하여 오차 포인트(outlier)를 제거하였다. 본 연구에서는 가상 격자와 kernel을 이용하여 각 군집의 표면 거칠기를 산출함으로써 수목을 인식하고, 포인트의 분리 과정 중에 1차원 필터링을 추가하여 건물의 수직벽에 반사된 포인트들을 군집화하고 제거하는 시도를 하였다. 보다 구체적이고 안정화된 방법론과 실험 결과에 대해서는 차후 논문에서 언급하고자 한다.

3. 적 용

선행 연구의 포인트 cloud의 분리 알고리즘(한수희 등, 2005, 2006)과 군집간 인접 관계 파악을 위해 보완된 알

표 3. 알고리즘의 처리 성능 비교

	처리 시간	메모리 사용
수정 전	2분 10초 내외	약 122MB
수정 후	2분 10초 내외	약 122MB

표 4. 샘플 자료와 처리 시스템의 제원

ALS 시스템	
모델명	ALTM 3070
제작사	Optech Inc.
총 포인트 수	3,066,991개
점밀도	약 1.5point/m <sup>2</sup> (단일 스트립)
처리 시스템	
CPU	AMD Athlon XP 2800+
RAM	1 GB
구현 언어	Visual C++ 6.0

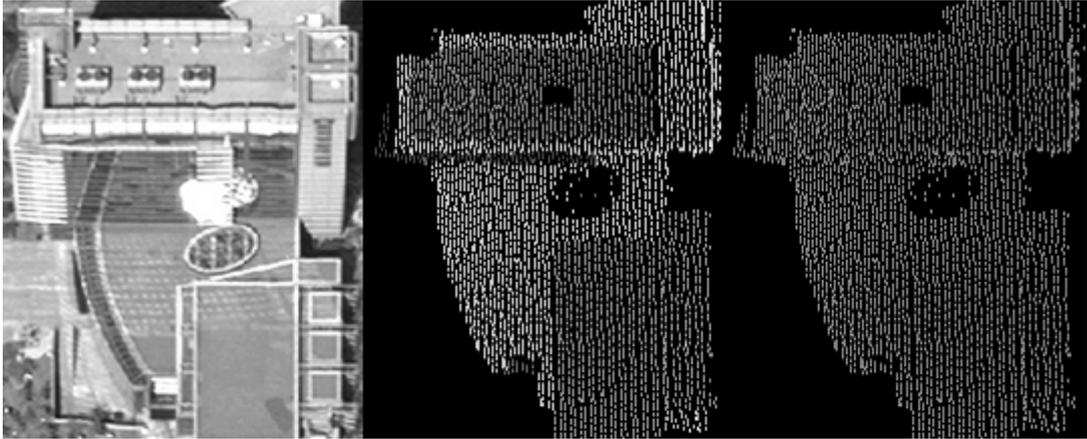


그림 11. 건물요소 인식과 결합의 예  
레퍼런스 영상(좌), 건물 요소의 인식(중), 건물 요소의 결합(우)



그림 12. 지면 포인트 군집

고리즘의 성능을 평가하기 위하여 단일 스트립에 대한 처리 시간과 순간 최대 메모리 사용량을 비교하였다. 결과적으로 처리 성능에는 거의 차이가 없음을 확인할 수 있었다(표 3). 사용된 샘플 자료와 처리 시스템의 제원은 (표 4)와 같다.

결과적으로 건물 요소로 분리되었던 아파트의 옥탑과 복잡한 형태의 지붕으로 구성된 건물의 건물 요소들을 결합하여 온전한 형태의 건물을 표현할 수 있었다. (그림 11)은 다양하고 복잡한 건물 요소로 구성된 샘플 건물의 처리 결과를 보여준다.

또한 지면 포인트 군집도 인접 군집 수가 평균 인접 군집 수를 압도적으로 초과하는 군집으로 선택함으로써 그 활용 가능성을 확인할 수 있었다(그림 12). 그러나 인공 구조물로 인해 고립되어 비교적 적은 수의 인접 관계를 갖는 소규모 지면 군집은 제시한 방식으로는 인식할 수 없는 한계점도 확인할 수 있었다.

#### 4. 결 론

본 연구에서는 선행 연구에서 제안한 스캔라인을 이용한 LIDAR 포인트 cloud의 분리과정에 분리된 포인트 군집간 인접 관계를 인식할 수 있는 기능을 추가하였다. 실험 결과, 새로운 기능의 추가로 인한 분리 과정의 처리 성능 저하는 발생하지 않았으며 분리된 군집간 인접 관계를 이용하여 건물 요소를 인식하고 재결합할 수 있었다. 또한 군집간 인접 관계를 지면 포인트 군집의 인식에서 활용할 수 있음을 확인하였다.

따라서 방대한 분량의 LIDAR 포인트 자료의 효율적이고 효과적인 처리는 물론 분리된 건물 요소의 인식과 결합을 통하여 건물 전체 형태 및 건물 요소의 개별 모델링을 위한 포인트 자료를 명확히 분리하여 제공할 수 있을 것으로 기대된다.

### 참고문헌

1. 한수희, 이정호, 유기윤, 2005, 항공레이저측량 자료의 스캔라인 특성을 활용한 건물 포인트 분리에 관한 연구, *한국지형공간정보학회논문집*, 제 13권, 4호, pp. 33-38.
2. 한수희, 이정호, 유기윤, 2006, ALS의 스캔라인 특성을 이용한 효율적인 포인트 클라우드의 분리, *공동춘계학술대회*, 2006.
3. Sithole, G., 2005, Segmentation and classification of airborne laser scanner data, Delft, Netherlands.
4. 이정호, 최재완, 유기윤, 2005, LIDAR 데이터의 스캔라인을 이용한 필터링, *공동춘계학술대회*, 2005.