**공간 데이터 관리 및 응용 Term Project document**

분산처리연구실

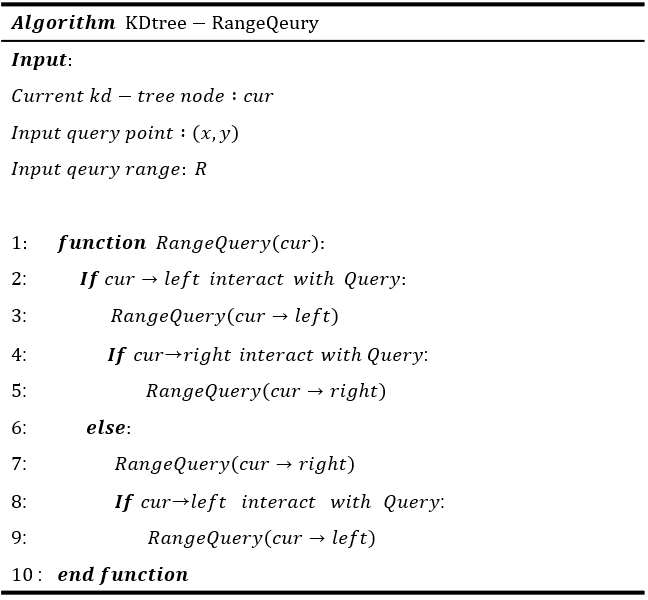
120180391 우상연

1. **프로젝트 목표**

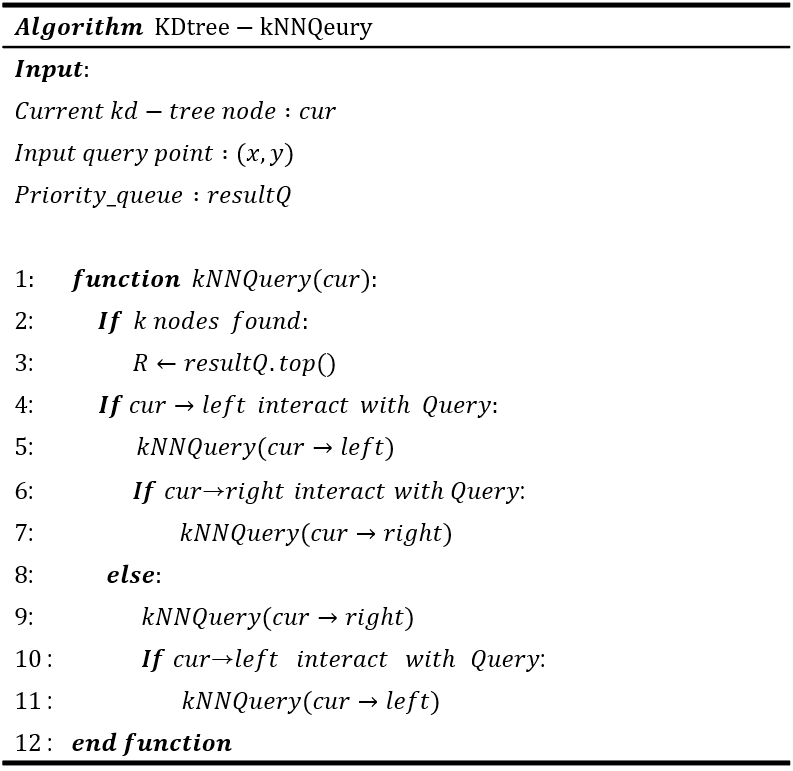
**KDtree**와 **Rtree**에 대해 공부하고 구현하여, 주어진 세가지 데이터에 대해 트리를 구성하고, 이 둘을 각각 point에 대한 질의에 대하여 **1.** brute-force algorithm, **2.** range query, **3.** kNN query를 구현하여 input값을 적절히 변화시키며 성능을 측정하였다.

1. **각 query별 pseudo code**

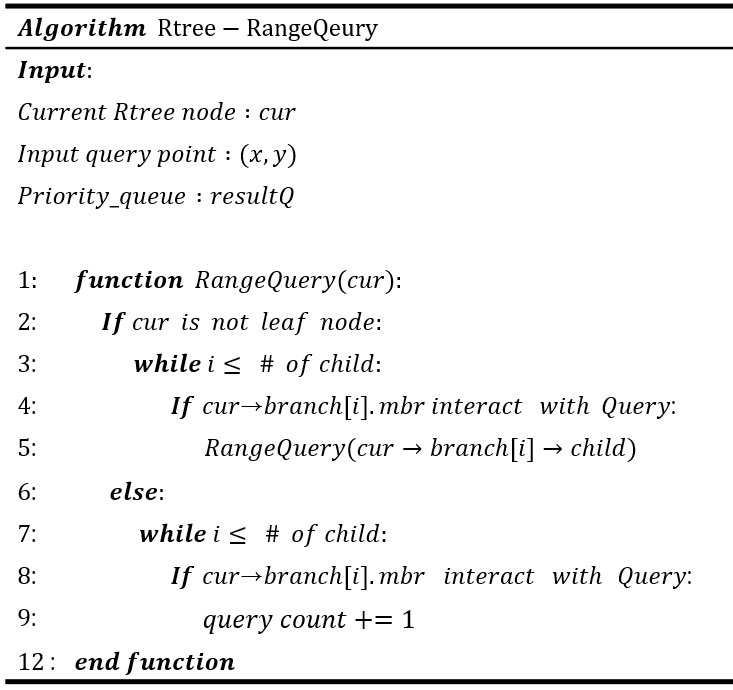
* **KDtree - RangeQeury**



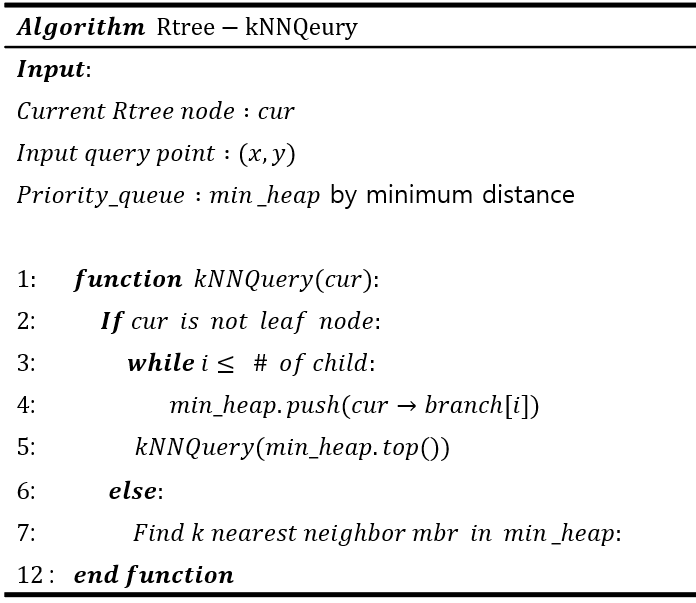
* **KDtree - kNNQeury**



* **Rtree – RangeQuery**



* **Rtree – kNNQuery**

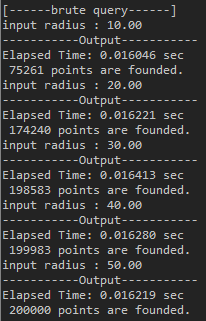
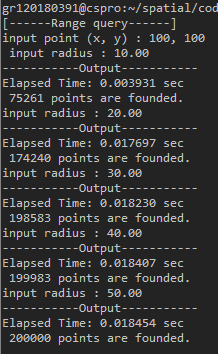


1. **성능 분석**

실험은 일정한 성능평가를 위해 학과 실습서버인 cspro.sogang.ac.kr에서 진행되었다.

* **실행결과 kdtree**

Range query와 성능비교를 위해 brute-force query는 kdtree를 DFS로 전 탐색 하는 방법으로 구현하였다. 아래 그림 1을 보면 같은 qeury에 대하여 brute force 방법과 range query의 point found결과가 동일함을 확인할 수 있다.



[그림 1]

kNN Query는 따로 결과값은 출력하지 않고 수행 시간을 측정하여 kdtree와 Rtree에서의 수행시간에 대해서 비교한다.

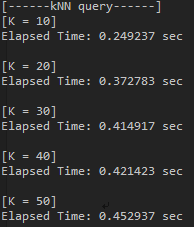
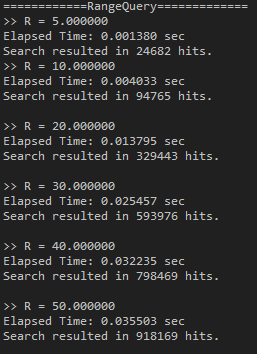
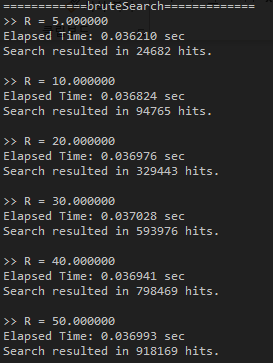
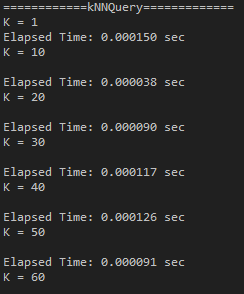


그림 2

* 실행결과 RTree
* 

Rtree 역시 수행결과 hit수가 동일함을 알 수 있다.  


**Clustered Dataset에 대한 성능비교**

**(x축 : radius, y축 : 걸린시간)**

그래프 1 kdtree 와 Rtree의 RangeQuery 성능비교

**(x축 : K, y축 : 걸린시간)**

그래프 2 kdtree 와 Rtree의 kNN쿼리 성능비교

**Uniformed Dataset에 대한 성능비교**

**(x축 : radius, y축 : 걸린시간)**

그래프 3 kdtree 와 Rtree의 RangeQuery 성능비교

**(x축 : K, y축 : 걸린시간)**

그래프 4 kdtree 와 Rtree의 kNN쿼리 성능비교

**Gaussian Dataset에 대한 성능비교**

**(x축 : radius, y축 : 걸린시간)**

**(x축 : K, y축 : 걸린시간)**

그래프 1, 2은 point(100,100)에 대하여 radius를 변화시켜가며 rangequery를 수행한 결과이다. 각 트리의 bruteforce방법은 단순히 array를 탐색하는 것이 아닌, 성능 비교를 위해 트리를 DFS방법으로 전탐색한 결과이다. (이는 range query에서 pruning되는 비교횟수에 대한 측정을 위함).

그래프 3, 4, 5, 6은 point(250,250)에 대해 radius를 변화시켜가며 rangequery를 수행한 결과이다.

Kdtree의 경우 작은 radius에서 bruteforce에 비해 좋은 성능을 보였지만, 그보다 큰 경우에서는 시간이 더 걸리게 되었는데, 이는 range계산 및 비교횟수가 더 많아진 이유이다. Rtree의 경우 대부분의 radius에서 brute force에 비해 성능이 좋았고, kdtree에 비해서도 월등한 성능을 보였다. 다만 radius가 100을 넘어가는 시점에서는 모든 node를 탐색하기에 kdtree보다 많은 노드를 가진 Rtree는 성능이 더 떨어지는 결과가 나왔다.

kNN쿼리의 경우 minimum distance를 활용한 Rtree의 방법이 kdtree에 비해 압도적으로 좋은 성능을 보였다. 이는 minimum distance를 통해 대부분의 MBR이 pruning되므로 비교횟수가 매우 줄어들기 때문으로 보인다.

이번 프로젝트에서 아쉬운 점은 dataset이 100만개로 매우 적은 점이다. Dataset이 매우 큰 데이터에 대해서는 range query가 bruteforce에 비해 훨신 성능 변화를 좋게 보여줄 수 있을 것 이다. 또한, range query와 kNN 쿼리 모두 Rtree가 좋은 성능을 보였고, 큰 range query에 대해서는 Rtree가 적합했으며, 두 종류의 트리 모두 대다수의 데이터를 포함하는 range에 대해서는 좋지않은 성능을 보였다.