건물 데이터베이스 구축을 위한 그래프 토폴로지 설계 및 패턴매칭 구현

Graph Topology Design for Generating Building Database and Implementation of Pattern Matching

최효석¹⁾ · 염재홍²⁾ · 이동천³⁾

Choi, Hyo-Seok · Yom, Jae-Hong · Lee, Dong-Cheon

Abstract

Research on developing algorithms for building modeling such as extracting outlines of the buildings and segmenting patches of the roofs using aerial images or LiDAR data are active. However, utilizing information from the building model is not well implemented yet. This study aims to propose a scheme for search identical or similar shape of buildings by utilizing graph topology pattern matching under the assumptions: (1) Buildings were modeled beforehand using imagery or LiDAR data, or (2) 3D building data from digital maps are available. Side walls, segmented roofs and footprints were represented as nodes, and relationships among the nodes were defined using graph topology. Topology graph database was generated and pattern matching was performed with buildings of various shapes. The results show that efficiency of the proposed method in terms of reliability of matching and database structure. In addition, flexibility in the search was achieved by altering conditions for the pattern matching. Furthermore, topology graph representation could be used as scale and rotation invariant shape descriptor.

Keywords : Graph database, Building topology, Shape descriptor, Pattern matching

초 록

3차원 건물을 모델링하기 위해 항공영상 또는 라이다 데이터를 이용하여 건물 외곽선 추출이나 지붕을 구성하는 패치를 추출하는 단계를 거친다. 이러한 3차원 정보를 자동으로 획득하는 알고리즘 개발과 같은 효과적인 정보의 획득에 대한 연구가 활발히 진행되고 있으나, 추후 추출된 정보의 활용이나 유지관리에 대한 연구는 미흡한 상태이다. 본 연구는 3차원 정보를 얻었다는 가정 하에 건물의 형태에 따른 검색을 위한 연구이다. 이를 위하여 벽면, 분할 지붕면, 바닥과 같은 건물의 구성체를 노드(node)로 표현하고 이들의 인접성 관계를 그래프 구조로 객체의 형태를 정의하는 토폴로지 설계 방법을 제안하였다. 제안된 방법에 의해 생성된 토폴로지를 건물 그래프 데이터베이스에 저장하고, 토폴로지 정보를 이용한 패턴매칭을 수행하여 건물을 검색한 결과의 분석을 통해 제안된 객체 토폴로지 설계방법의 효용성을 입증하였다. 그래프 구조의 토폴로지를 기반으로 건물을 검색할 수 있었으며, 검색 조건을 부여하여 건물의 유사 정도를 조절하며 검색할 수 있었다. 또한 축척 및 회전에 불변한 객체의 형태모사 방법으로 사용될 수 있다고 사료된다.

핵심어 : 그래프 데이터베이스, 건물 위상구조, 형태모사, 형태정합

1) Dept. of Geoinformation Engineering, Sejong University, Korea(E-mail: seok@sju.ac.kr)

2) Corresponding author · Member · Dept. of Geoinformation Engineering, Sejong University, Korea(E-mail: jhyom@sejong.ac.kr)

3) Member · Dept. of Geoinformation Engineering, Sejong University, Korea(E-mail: dclee@sejong.ac.kr)

1. 서론

항공영상 또는 라이다 데이터를 이용한 건물의 외곽선 추출이나 지붕 패치를 추출하는 등 데이터로부터 3차원 정보를 자동으로 얻기 위한 알고리즘에 대한 연구가 진행되고 있다(Kim and Kim, 2012; Kim and Lee, 2009). 또한, Joo *et al.*(2012)은 효과적인 3차원 공간정보 구축에 대한 평가 방안을 제시하였다. 3차원 공간정보 구축이나 평가에 대한 연구는 활발하지만 구축된 데이터의 활용에 대한 연구는 미흡하다. 본 논문은 원시 데이터(예: 항공 라이다 데이터 또는 영상)로부터 건물의 3차원 정보를 얻었다는 가정 하에 또는 수치지도 등 기 구축된 3차원 건물 데이터를 이용하여 건물의 형태에 따른 동일 또는 유사 건물객체의 효율적인 검색에 관한 연구이다.

건물을 검색하는 방법으로는 아이디어나, 주소 등 건물 객체에 입력된 속성을 이용하여 검색을 할 수 있으며, 입력된 속성에 따라 다양한 검색을 할 수 있으나 속성을 이용하여 건물의 형태를 표현하거나 검색하는 것은 부적합하다. Verma *et al.*(2006)은 항공 라이다 데이터를 이용하여 복잡한 도심지의 3차원 모델링과 객체의 기하학적 구조를 생성하였다. 특히, 건물의 지붕면을 추출하여 지붕면들 간의 인접성과 인접하는 각도를 이용하여 토폴로지를 구성하고 그래프 구조를 이용하여 표현함으로써 그래프의 효율성을 제시하였다. 이는 복잡한 구조를 갖는 데이터를 저장하는데 구조를 직관적으로 표현할 수 있는 그래프를 이용한 저장방식이 효율적이라 판단되고, 저장된 데이터를 이용하는 방법이 필요하다.

본 연구의 목적은 그래프 구조를 이용하여 건물의 토폴로지를 설계하는 방법을 제안하고, 토폴로지 패턴매칭을 통하여 완전히 일치하는 동일 형태의 건물이나 유사한 건물을 검색하는 수행하여 제안한 방법의 효율성을 평가하는 것이다. 실험을 위해 제안한 방법으로 설계한 토폴로지 그래프를 이용하여 다양한 형태의 건물 데이터를 생성하고 데이터베이스화 하였다. 그래프 구조로 구축된 건물의 토폴로지를 패턴매칭을 통해 검색을 수행하고 결과를 분석하였다. 이를 위하여 JAVA 1.6.0_33으로 프로그래밍하였으며, 그래프 데이터베이스는 Neo4j-1.8을 사용하여 구축하였다. 웹서버는 Apache-Tomcat 6.0.35를 사용하였고, JIT(JavaScript InfoVis Toolkit)를 이용하여 검색 결과를 시각화 하였다.

2. 건물의 그래프 토폴로지 데이터베이스 구축

2.1 그래프 데이터베이스의 개요

최근 클라우드 컴퓨팅과 SNS의 발달 등으로 저장할 데이터양과 종류가 기하급수적으로 증가하고 있다. Manyika *et al.*(2011)은 빅데이터를 기존 데이터베이스의 수집·저장·관리·분석의 범위를 넘어서는 대량의 데이터셋이라 정의하였고 Gantz and Reinsel(2011)은 이러한 데이터로부터 새로운 가치를 도출하고 결과를 분석하는 기술을 의미한다고 하였다. 이와 같이 방대하고 다양한 데이터를 입력하고 처리하기 위하여 기존에 많이 사용되고 있는 관계형 데이터베이스 구조는 여러 문제점을 야기하고 한계에 도달하게 될 것이다. 이와 같은 현상은 빅데이터 시대에 당면한 문제이며, 데이터양과 종류 및 형태가 방대한 공간정보 데이터 처리를 위하여 해결해야 할 과제이다.

빅데이터에 적합하지 않는 관계형 데이터베이스의 대안으로 NOSQL(Not Only SQL)이라는 새로운 개념의 데이터베이스가 등장하였다. NOSQL의 장점은 비관계형으로 관계형 데이터베이스의 테이블 사이를 연결하는 JOIN을 사용하지 않기 때문에 검색 속도가 빠르다. 또한 테이블과 같은 일정하고 획일화된 구조를 가지고 있지 않으므로 유연하게 확장성이 용이하고, 단순한 key와 value의 쌍으로 데이터를 저장하기 때문에 대용량 데이터를 처리하는데 적합하다.

NOSQL은 저장하는 방식에 따라 다양한 데이터모델을 가지고 있으며, Key-Value, Column family, Document database 및 Graph database로 분류된다. 특히, 그래프 데이터베이스는 그래프의 노드(Node)와 관계(Relationship), 속성(Property) 구조로 데이터를 저장한다. 이러한 그래프 구조는 어떠한 현상도 직관적으로 저장할 수 있고, 그래프 구조에 따라 다양한 패턴들이 생성되고, 패턴매칭에 의한 네트워크 분석이 용이한 장점이 있다.

2.2 그래프 구조의 토폴로지 설계 방법

건물의 형태는 다양하고 복잡하지만 건물을 구성하는 기본 요소는 면이고, 건물은 이러한 면들의 조합으로 이루어져 있다. 대부분의 건물의 벽면은 단일 평면이지만 지붕은 여러 평면의 조합뿐 아니라 구 또는 원통과 같이 곡면인 경우도 있다. 그래프 데이터베이스에서 노드는 하나의 단위 면을 표현할 수 있고 면들 간의 관계는 노드와 인접노드 사이의 관계를 정의할 수 있다. 본 논문에서는 노드를 이용하여 모든 면을 정의하고 관계를 이용하여 면 사이의 인접성을 결정하였다. 또한 노드와 관계에 대한 속성을 이용하여 부가적인 정보를 추가하였다.

면과 면간의 인접성으로 구성된 그래프 구조는 그래프의 패턴을 기반으로 검색을 수행할 수 있으며, 부가적으로 속성

을 이용한 검색도 가능하다. 또한 두 가지 조합을 이용하여 복잡한 형태의 객체를 검색하는 것도 가능하다. 이러한 인접성과 속성을 이용하여 검색을 하기 위해서는 정해진 규칙에 의해 데이터베이스가 구축되어야 한다. 규칙에 의해 정의된 건물의 토폴로지를 이용하여 검색뿐 아니라 원래의 건물 형태도 복원할 수 있어야 한다.

본 연구에서 정한 규칙은 노드 자체와 노드간의 관계에 대한 규칙으로 구별하여 결정하였다. 건물의 면을 의미하는 노드는 면의 분류, 모양, 변의 개수를 고려하였으며, 면들 간의 인접성을 결정하는 관계는 인접하는 방향과 각도를 고려하였다. 또한 건물의 크기와 회전에 상관없이 건물의 유사성을 판단하여 검색하는 것을 목적으로 하기 때문에 크기와 회전에 관한 요소는 고려할 필요가 없다. 그러므로 그래프 구조의 토폴로지는 축적과 회전에 불변한 객체표현 방법인 Scale and rotation-invariant shape descriptor이다.

2.3 토폴로지 규칙

Fig. 1의 그래프에서 숫자는 면이고, 화살표는 인접성을 표현하며, 단순한 형태의 하나인 육면체 객체의 그래프 토폴로지를 보여주고 있다. 관계는 면과 면이 변으로 인접하여야 한다. 직육면체 건물의 경우 면이 6개가 존재하고, 각 면마다 4개의 관계가 형성된다. 그러나 복잡하고 다양한 형태의 건물에 대한 토폴로지를 구성하기 위해서는 규칙이 필요하다. 토폴로지 설계방법에서 설명한 바와 같이 이러한 규칙은 모든 형태의 건물에 적용이 될 수 있어야 하며 규칙에 의해 정의된 토폴로지와 규칙을 이용하여 다시 원래 형태의 건물로 복원할 수 있어야 한다.

Fig. 1. Object and graph topology

Table 1에서 보여주는 것처럼 노드와 관계를 구분하여 규칙을 생성하였으며, 노드는 면의 종류(Type), 모양(Shape), 변의 수(Number of sides)로 분류하였고, 관계는 면의 인접방향관계(Orientation)와 인접한 면간의 각도(Angle)를 분류하였다.

Table 1. Graph topology design

Graph	Category	Element
Node	Type	Roof
		Side wall
		Floor
	Shape	Plane
		Cylinder
		Sphere
	Number of sides	3
		4
		...
Relationship	Orientation	Up
		Side
		Front
	Angle	0°~ 90°
		90°
		90°~ 180°
		180°
		180°~ 270°
		270°
		270°~ 360°

(1) 노드 규칙

노드 규칙은 단위가 면인 노드의 속성에 기준을 부여하는 것이다. 본 연구에서는 건물의 외부 구조를 다루고 있기 때문에 건물 내부의 면들은 고려하지 않았다. Fig. 2(a)는 건물 면의 종류(Type)를 지붕(Roof), 벽면(Side wall) 그리고 연결성을 고려하기 위하여 바닥면(Floor)을 추가한 것을 보여주고 있다. 노드에 대한 규칙을 정의하는 것은 검색을 위한 패턴매칭의 필터링에서 핵심적인 역할을 한다.

Fig. 2(b)는 면의 모양(Shape)을 분류한 것이다. 건물을 이루는 면은 대부분 평면(Flat 또는 Slant plane)의 형태이지만 곡면인 원통(Half-cylinder)이나 반구(Hemisphere 또는 Dome) 형태가 일반적이다. 각각의 면의 모양에 따라 plane, cylinder, sphere로 정의 하였다. 대부분의 면은 평면이기 때문에 평면인 노드는 속성 자체를 정의하지 않았으며, 검색 결과가 Null값이 되도록 하였다. 이는 저장 공간을 절약할 수 있다. Fig. 2(c) 변의 수를 분류한 것이다. 변의 수는 면으로 구성된 객체가 포함하는 변의 개수이다. 하나의 면은 3개 이상의 변으로 구성되어야 하므로 속성값은 3이상인 값을 갖는다.

Fig. 2. Node rules

(2) 관계 규칙

관계 규칙은 인접한 면들 간의 상태에 대한 속성을 부여하는 것으로서 방향과 각도로 정의된다. Fig. 3(a)는 면과 면 간의 방향(direction)에 따라 분류한 것이다. 바닥에서 벽, 벽에서 지붕같이 아래에서 위로 연결된 관계는 “Up”으로 정의하였고, 벽에서 벽, 지붕에서 지붕같이 옆으로 연결된 연결은 “Side”로 정의하였다. 또한 지붕면들이 변으로 만나고 마주 보고 있는 경우 “Front”로 정의하였다.

Fig. 3(b)는 면과 면이 이루는 각도(angle)에 따라 분류한 것이다. 건물의 안쪽으로 이루는 각을 기준으로 하여 7가지($0^\circ \sim 90^\circ$, 90° , $90^\circ \sim 180^\circ$, 180° , $180^\circ \sim 270^\circ$, 270° , $270^\circ \sim 360^\circ$)로 분류하였다. 일반적으로 대부분의 건물들은 서로 직각을 이루기 때문에 90° , 180° , 270° 를 기본 각도로 정하고 그 사이의 범위로 지정해 주었다. 각도의 범위를 세밀하게 분할할수록 건물의 형태를 정밀하게 표현하여 객체의 묘사 세밀도(Level-of-Detail)를 향상할 수 있지만 데이터 베이스 구축에 시간과 비용이 소비될 수 있다. 그러나 정형화된 건물 객체의 검색은 다른 조건들이 복합적으로 사용되므로 각도를 세분화하는 것은 효용성 측면에서는 큰 효과가 없다.

Fig. 3. Relationship rules

Fig. 4. Test building models and graph topology structures

3. 패턴매칭에 의한 건물 검색 실험

3.1 테스트 건물모델 생성

본 연구에서 제한한 건물 토폴로지 설계방법을 검증하고

평가하기 위하여 전형적이고 대표적인 건물 형태를 3가지 종류로 구분하여 테스트 건물모델을 생성하였다. Fig. 4에서 보여주는 것처럼, 건물모델 (a), (b), (c)는 지붕면의 형태는 다르지만 동일한 그래프 구조를 갖는 건물을 생성하였고, (d), (e), (f)는 다양한 형태의 지붕으로 구성된 건물이다. 또한 (g), (h), (i)는 복잡한 그래프 구조를 갖는 건물모델을 생성하였다. 또한 Fig. 4는 각 건물모델에 대한 그래프 구조를 보여주고 있다. 이와 같은 건물은 수치지도와 같은 기존의 3차원 건물 데이터 또는 다양한 데이터(예: 항공 라이다 및 항공영상)를 이용하여 생성된 결과를 활용할 수 있다. 서론에서 언급한 바와 같이 본 연구는 건물 객체의 생성이 목적이 아니고 기 생성된 데이터를 이용한 건물 데이터베이스 구축의 새로운 방법을 제시하는 것이다.

3.2 건물 데이터베이스 구축

오픈소스 그래프 데이터베이스인 Neo4j를 사용하여 건물 데이터베이스를 구축하였다. 데이터베이스에 입력된 테스트 데이터는 노드 112개, 관계 218개 및 속성 348개로 구성되어 있다. 구축한 데이터베이스는 실시간으로 접근과 공유가 가능하므로 다양한 애플리케이션에서 활용할 수 있다. Neo4j는 그래프 구조기반의 패턴매칭을 통해 정보를 추출하는 Cypher를 제공한다. Cypher는 그래프 쿼리(Query) 언어로서 노드와 관계 구조를 이용하여 검색하고 추가적으로 속성을 이용하여 검색을 할 수 있다. 본 연구에서는 인터넷 브라우저인 Chrome에서 Cypher 쿼리를 입력하고 구축된 데이터베이스에 접근하여 브라우저를 통해 시각화된 결과를 확인할 수 있도록 구현하였다.

Cypher는 Table 2에 명시된 구문을 이용한다. START는 패턴매칭이 시작되는 노드나 관계를 의미한다. MATCH는 그래프의 구조를 정의하여 패턴매칭을 하는 부분이고, WHERE는 노드와 관계의 속성을 이용하여 필터링 하는 부분이다. RETURN은 반환하는 결과값의 형태를 정의한다.

Table 2. Syntax of Cypher

Clauses	Explanation
START	Node or relationship of starting point
MATCH	Specifying the graph pattern
WHERE	Filtering apart from the pattern of the data
RETURN	Defining the type of the results

3.3 검색 수행

구축된 데이터베이스에서 검색하고자 하는 건물 형태의 그래프를 입력하고, 유사성이 높은 데이터를 검색하는 방법으로 실험을 수행하였다. 앞에서 정의된 규칙을 기초로 하여 검색하고자 하는 건물의 그래프 구조를 Cypher를 이용하여 입력해야 한다.

(1) 실험 1: Gable 지붕

실험 1은 단순 Gable 형태의 지붕(Fig. 4(f))을 갖는 패턴을 데이터베이스에서 검색하는 것을 수행하였다. ORDER BY는 오름차순으로 결과를 출력하는 것이다. Gable 지붕은 MATCH에서 바닥인 e를 기준으로 벽 a, d가 인접해있고 각각 지붕 b와 c가 인접하는 형태로 그래프의 패턴을 정의하였으며, 각 면의 종류와 벽과 지붕이 이루는 각도가 90°~180°인 필터링 과정을 WHERE에서 정하였다.

< Query for Test 1 >

START a = node(*)

MATCH

p = e-->a-[r1]->b-[r2]->c<-[r3]-d<--e

WHERE a.type! = "wall" and b.type! = "roof" and c.type! = "roof" and d.type! = "wall" and r2.direction! = "front" and e.type! = "floor" and r1.angle! = "90-180" and r2.angle! = "90" and r3.angle! = "90-180"

RETURN e.b_id, e.b_name, count(e.b_id)

ORDER BY e.b_name;

Table 3는 쿼리를 실행한 결과이며, 건물 id와 건물 이름, 검색된 건물의 수를 나타내고 있다. Gable과 같은 특성을 가지고 있는 건물은 Fig. 4의 (f), (g), (h) 및 (i)이다. 건물 7은 Gable 지붕 형태 2개를 가지고 있으며, 건물 8과 건물 9는 각각 4개의 Gable 지붕으로 구성되어 있다.

Table 3. Results from Test 1

e.b_id	b.b_name	count(e.b_id)
6	Building 6	1
7	Building 7	2
8	Building 8	4
9	Building 9	4

Fig. 5는 검색을 실행한 결과에 대한 그래프 구조를 시각화 한 것이며, 검색된 결과가 동일한 그래프 패턴을 보이는 것을 확인할 수 있다. Fig. 6은 패턴매칭에 의해 검색된 건물과 Gable 지붕의 개수를 보여주고 있다.

Fig. 5. Results from Test 1

Fig. 6. Matching buildings from Test 1

(2) 실험 2: L형태의 Gable 지붕

실험 2는 Gable 형태의 지붕이 두 개가 직각으로 연결되어 있는 Fig. 4(g)와 같은 형태를 데이터베이스에서 검색하는 것이다. Gable 지붕 두 개가 직각으로 만나는 특징을 이용하여 검색을 수행하였다. 쿼리의 MATCH에서 a, b, c 및 d는 지붕면이고, r1, r2, r3 및 r4는 면의 인접을 나타낸다. 또한 WHERE에서는 필터링을 위한 요소를 결정하게 된다. 지붕의 인접성을 이용하여 검색을 하였으며, 데이터 저장의 중복을 막기 위해서 건물 id와 건물 이름은 건물의 바닥의 속성에만 저장

되어있기 때문에 이 속성을 가져오기 위하여 바닥을 연결하는 패턴을 추가하고 검색 결과를 출력을 하였다.

< Query for Test 2 >

START a = node(*)

MATCH

p = a-[r1]-b-[r2]->c-[r3]-d-[r4]->a,
e-->f-->a

WHERE a.type! = "roof" and b.type! = "roof" and c.type!
= "roof" and d.type! = "roof" and r1.direction! = "front"
and r2.direction! = "side" and r3.direction! = "front" and
r4.direction! = "side" and r2.angle = "270" and r4.angle =
"90" and e.type = "floor"

RETURN e.b_id, e.b_name, count(e.b_id)

ORDER BY e.b_name;

Table 4는 실험 2의 검색 결과이고 건물 3과 같은 지붕의 패턴을 보이는 건물은 건물 7과 건물 8이다. 건물 7은 검색한 대상이 검색된 것이고 건물 8는 건물 7의 패턴을 4개를 가지고 있다.

Table 4. Results from Test 2

e.b_id	b.b_name	count(e.b_id)
7	Building 7	1
8	Building 8	4

Fig. 7은 검색된 그래프 구조를 보여주고 있으며 Fig. 7(a)는 건물 7이고, Fig. 7(b)는 건물 8의 그래프 구조의 일부를 시각화한 것이며, Gable 지붕의 그래프 구조와 일치함을 알 수 있다. 건물 8의 경우는 L형태의 Gable 지붕이 네번 중복되어서 표현이 되었다. Fig. 8은 패턴매칭에 의해 검색된 건물의 Gable 지붕 개수를 보여주고 있다.

Fig. 7. Results from Test 2

Fig. 8. Matching buildings from Test 2

(3) 실험 3: 단면지붕

실험 3은 그래프의 구조는 같으나 건물의 형태는 다른 경우를 검색하는 것이다. 건물 1은 모든 면이 평면이고 6개의 면으로 이루어진 직육면체 건물이다. 건물 2는 직육면체 건물이나 지붕이 기울어진 형태이고, 건물 3은 직육면체 건물과 유사하나 지붕이 구면인 건물이다. Fig. 4에서 세 건물의 그래프 구조를 보면 같은 구조를 가지고 있다.

아래는 실험 3의 쿼리이다. 속성을 이용하지 않고 그래프 구조만을 이용하여 육면체 건물을 검색하였다. 쿼리의 MATCH에서 p1은 바닥에서 벽, p2는 벽에서 벽, p3는 벽에서 지붕의 인접성을 나타낸다.

< Query for Test 3 >

START a = node(*)

MATCH p1 = a-->b, a-->c, a-->d, a-->e, p2 = b-->c-->d-->e-->b, p3 = b-->f, c-->f, d-->f, e-->f

RETURN a.b_id, a.b_name, f.shape;

Fig. 9에서 동일한 그래프 구조가 두개의 건물에서 나타나는 것을 확인 할 수 있다. 검색 결과는 Table 5이며 건물 1, 건물 2 및 건물 3이 검색된 것을 확인할 수 있다. 각 건물이 4번씩 검색된 것은 4개의 벽에서 각각 동일한 패턴으로 나타나기 때문이다. 또한 건물 1과 건물 2의 shape가 null인 이유는 데이터베이스에서 shape의 flat(평면)은 속성을 입력하지 않았기 때문이다. 건물 1과 건물 2는 shape라는 속성 자체를 가지고 있지 않다. 그러므로 쿼리의 RETURN에서 느낌표를 붙여주고 속성이 없는 결과는 null값을 출력하도록 하였다. 이는 데이터를 생성할 때 불필요한 작업을 줄이고 데이터의 저장 공간을 절약할 수 있는 NOSQL의 특징이다. Fig. 10은 패턴매칭에 의해 검색된 건물을 보여주고 있다.

Table 5. Results from Test 3

a.b_id	a.b_name	f.shape!
1	Building 1	null
1	Building 1	null
1	Building 1	null
1	Building 1	null
2	Building 2	null
2	Building 2	null
2	Building 2	null
2	Building 2	null
3	Building 3	sphere
3	Building 3	sphere
3	Building 3	sphere
3	Building 3	sphere

Fig. 9. Results from Test 3

Fig. 10. Matching buildings from Test 3

동일한 형태의 그래프 구조로 구성된 건물 1, 건물 2 건물 3을 구별하기 위해서는 속성을 사용하여 검색을 하여야 한다. 먼저 건물 2의 특징은 기울어진 지붕이다. 건물 2의 기울어진 지붕은 벽과 지붕이 만나는 관계의 angle속성을 이용하여 정의한 것이다. 이것을 이용하여 MATCH에 관계의 속성을 이용할 수 있도록 r1과 r2를 추가하고 WHERE에 angle의 조건을 추가하여 준다. 다음 구문을 쿼리에 추가시키면 된다.

< Adding Angle Condition >

MATCH p1 = a-->b, a-->c, a-->d, a-->e, p2 = b-->c-->d-->e-->b, p3 = b-->f, c-[r1]->f, d-->f, e-[r2]->f

WHERE

r1.angle! = "90-180" and r2.angle! = "0-90"

Table 6은 추가하여 수정된 쿼리를 이용한 실행 결과이며, 이 경우 경사 지붕인 건물 2가 검색되었다. 검색된 결과가 하나인 이유는 벽에서 지붕으로 인접하는 각도를 이용하여 검색을 수행하였기 때문에 벽마다 같은 패턴이 발생하지 않기 때문이다.

Table 6. Result from adding angle

a.b_id	a.b_name	f.shape!
2	Building 2	null

건물 3의 차이는 지붕의 속성 shape값이 있는 것이다. 건물 1과 건물 2는 편평하기 때문에 다폴트인 flat으로 shape라는 속성을 가지고 있지 않고 건물 3은 구형 지붕으로 지붕의 shape속성이 sphere이다. 이것을 이용하여 검색을 하기 위해서는 WHERE에 지붕의 shape속성이 sphere라는 조건을 입력하여야 한다. 다음 구문을 쿼리에 추가시키면 된다.

< Adding Sphere Shape Condition >

WHERE f.shape! = "sphere"

Table 7는 수정한 쿼리의 실행결과이며, 구형 지붕(Dome) 형태의 건물 3이 검색되었다. 그러므로 패턴매칭의 검색 방법을 통해서 완전히 일치하는 데이터를 검색할 수 있을 뿐만 아니라 검색하는 조건에 따라 검색의 정확도를 결정할 수 있었다.

Table 7. Result from adding sphere shape

a.b_id	a.b_name	f.shape!
3	Building 3	sphere
3	Building 3	sphere
3	Building 3	sphere
3	Building 3	sphere

4. 결 론

본 연구는 노드와 관계를 이용한 그래프 구조를 기반으로 건물의 토폴로지를 설계하는 방법을 제안하였다. 또한 토폴로지 패턴매칭을 수행하여 건물을 검색하는 실험을 통하여 다음과 같은 결론을 얻을 수 있었다.

- (1) 면 사이의 인접을 정의한 그래프는 패턴매칭의 기초가 되고, 노드와 관계의 속성은 검색을 구체화하며 최소한의 규칙을 이용하여 검색하는 것이 효율적인 방법이다.
- (2) 패턴매칭을 통한 검색은 건물 객체의 회전과 축척에 영향을 받지 않는 장점이 있다. 또한 그래프데이터베이스는 관계형 데이터베이스와는 다르게 스키마가 정의되어 있지 않기 때문에 속성 추가가 자유롭다.
- (3) 실험 1 : 복잡한 형태의 건물의 일부를 검색할 수 있었다. 완벽히 일치하는 객체를 검색하는 것이 아니고 형태의 일부를 포함되는 건물을 검색하는 것이기 때문에 데이터베이스 전체에 적용될 수 있도록 검색 조건을 부여하여야 한다.
- (4) 실험 2 : 지붕의 형태만을 이용하여 검색을 수행 하였다. 건물 전체의 토폴로지를 이용하지 않고 지붕의 토폴로지만을 이용함으로써 지붕 형태가 유사한 건물을 검색할 수 있었다.
- (5) 실험 3 : 건물의 토폴로지 구조는 동일하지만 건물의 형태가 다른 경우의 건물을 검색하였다. 그래프 구조는 같으나 속성을 이용한 검색조건에 따라 다른 검색 결과를 얻었다. 그러므로 검색에 사용하는 속성에 따라 유사 정도를 조절할 수 있었다.
- (6) 제시된 토폴로지 설계방법에 의해 실제 건물의 데이터베이스를 구축하면 그래프의 유사정도를 조절하며 검색을 할 수 있지만, 실제 건물의 경우 테스트 건물모델에 비해 형태가 복잡하기 때문에 데이터베이스를 구축하는 단계에서 시간이 소요된다. 추후 이러한 단계를 자동화할 수 있는 연구가 필요하다.

감사의 글

이 논문은 2013년도 정부(교육부)의 재원으로 한국연구재단의 기초연구사업(No. 2011-0012868)과 공간정보 전문인력 양성사업의 지원을 받아 수행된 것입니다.

References

- Gantz, J. and Reinsel, D. (2011), *Extracting Value from Chaos*, International Data Corporation(IDC), IDC 1142, Framingham, U.S.A., pp. 1-12.
- Joo, Y., Kim, K., and Hahm, C. (2012), An Evaluation Scheme on Feasibility in Public Sector for 3D Geo-Spatial Information - Focusing on Production of Digital Mapping, *Journal of The Korean Society for GeoSpatial Information System*, Vol. 20, No. 3, pp. 73-82. (in Korean with English abstract)
- Kim, N. and Kim, Y. (2012), A building outline extraction scheme using tile-based topographical classification from aerial LiDAR data and building tile's airborne image, *Korea Computer Congress 2012*, KIISE, 27-29 June, Jeju, Korea, Vol. 39, pp. 4-6.
- Kim, S. and Lee, I. (2009), Regularization of 3D Building Models, *KSRS Spring Conference 2009*, KSRS, 29 March, Seoul, Korea, pp. 296-300.
- Manyika, J., Chui, M., Brown, B., Bughin, J., Dobbs, R., Roxburgh, C. and Byers, A. H. (2011), *Big data: The Next Frontier for Innovation, Competition, and Productivity*, McKinsey Global Institute, New York, USA, pp. 1.
- Verma, V., Kumar, R. and Hsu, S. (2006), 3D Bilding Detection and Modeling from Aerial LiDAR Data, *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, New York, 17-22 June, USA, Vol. 2, pp. 2213-2220.