

논문 2023-3-6 <http://dx.doi.org/10.29056/jsav.2023.09.06>

부분지도 삭제 객체 탐지를 위한 부분 변경 객체 검출 방법 비교 연구

김광수*, 김봉완*†

A Comparative Study on Methods for Identifying Partially Changed Objects to Detect Deleted Objects in Partial Maps

Kwangsoo Kim*, Bong Wan Kim*†

요 약

도심지 지하공간 안전사고의 예방을 위해 지하공간통합지도가 구축되고 있다. 지하공간통합지도의 갱신 시간을 단축시키기 위해 신규로 입력된 지도 중 갱신된 객체만을 선별하는 변화객체 탐지추출 기술이 적용되었다. 갱신 객체는 신규 객체, 삭제 객체, 속성 변경 객체로 분류한다. 그러나, 굴착 공사의 대상이 되는 매설물만 포함한 부분지도가 입력된 경우 삭제 객체를 찾지 못하는 문제가 발생한다. 본 논문에서는 부분지도에서 삭제 객체를 찾을 수 있는 세 가지 방법들에 대해 논의하였다. 삭제 객체를 찾기 위해서는 원본 객체의 일부만 변경된 갱신 객체의 존재 여부를 판정하여야 한다. 일부가 변경된 객체를 추출하기 위해 DE-9IM, Relate 함수, Intersection 함수를 사용하였으며, 이 방법들의 성능을 선형 객체에 적용하여 비교하였다. 세 가지 방법 모두 중첩된 형상을 포함한 객체들을 동일 객체로 오류 없이 판정하였으며, Relate 함수가 가장 빠른 성능을 보여주었다.

Abstract

In order to prevent safety accidents in urban underground spaces, an integrated underground space map is being generated. To shorten the update time of the map, a changed object detection and extraction technology is applied, which selects only updated objects from newly input maps. Updated objects are classified into a new object, a deleted object, and an attribute changed object. However, when a partial map including only buried facilities that are targets of excavation work is input, a problem of not finding a deleted object occurs. In this paper, we discussed three methods to find deleted objects in the partial map. In order to detect the deleted object, it must be determined whether an updated object exists in which only part of the original objects has been changed. DE-9IM, Relate function, and Intersection function were used to extract partially changed objects, and the performances of these methods were compared by using linear objects. All three methods determined objects including overlapping shapes as the same object without errors, and the Relate function showed the fastest performance.

한글키워드 : 지하공간, 통합지도, 부분지도, 삭제 객체, 부분 변경 객체

keywords : underground, integrated map, partial map, deleted objects, partially changed object

* 한국전자통신연구원 도시·공간ICT연구실

접수일자: 2023.09.12. 심사완료: 2023.09.20.

† 교신저자: 김봉완(email: kimbw@etri.re.kr)

게재확정: 2023.09.20.

1. 서론

도시로의 인구 집중은 필연적으로 사용 가능한 토지의 부족, 인구 과밀, 교통 혼잡 등의 문제를 발생시킨다. 이러한 문제를 해결하기 위해 지하공간에 대한 개발이 진행되었고, 지하에는 상수도, 하수도, 통신, 가스, 교통, 저장, 상업 시설 등이 건설되어 사용 중이다. 지하공간을 개발하기 위해서는 반드시 굴착이 이루어진다. 그러나, 한번 파헤쳐진 지하공간은 원래의 균형 잡힌 상태로 매립하는 것이 거의 불가능하며, 균형을 상실한 지하의 상태는 다양한 문제를 언제든지 발생시킬 수 있는 위험을 내포하고 있다. 이렇게 균형이 깨진 지하의 상태와 노후화된 지하매설물의 파손 등에 기인한 지하 안전사고가 서울과 같은 대도시를 중심으로 지속적으로 증가하고 있다. 그림 1은 2018년부터 2022년사이 5년 동안의 지반 함몰 발생 현황을 나타내고 있으며, 연평균 225건의 사고가 발생하였다[1].



그림 1. 연도별 지반 함몰 발생 건수
Fig. 1. number of ground cave-ins by year

빈번하게 발생하는 지하 안전사고는 시민들에게 불안감을 느끼게 한다. 수도권 성인 1000명을 대상으로 한 설문 조사에서 95% 이상의 시민이 지반 함몰에 대한 불안감을 느끼고 있는 것으로 조사되었다[2]. 정부에서는 지하 안전사고에 대한 시민의 불안감 해소 및 안전 확보를 위한 대책으로 지하공간의 효율적인 관리를 지원하기 위해 국

토교통부를 중심으로 2015년부터 3차원 기반 지하공간통합지도 구축 사업을 진행하고 있다[3]. 지하공간통합지도는 지하에 매설된 지하시설물 6종과 지하구조물 6종 및 지반정보 3종으로 구성되어 있다. 지하시설물은 상수, 하수, 통신, 가스, 난방, 전력 등 관로형의 시설물이며, 지하구조물은 지하철 역사 및 선로, 지하차도, 지하보도, 지하주차장, 지하상가, 공동구 등 구조물형의 시설물이며, 지반은 시추, 관정, 지질 등을 의미한다.

지하공간통합지도는 지하 개발 사업에 직접 활용하기 위해 최신 정보 유지가 필요하다. 그러나, 변경된 지하정보가 실시간으로 지하공간통합지도에 반영되지 못하는 문제가 있다. 이러한 문제를 해결하기 위해 지하공간통합지도 실시간 갱신체계 구축을 위한 기술 개발이 진행 중이며, 실시간 갱신체계 구축의 일환으로 변화객체 탐지추출 기술이 개발되고 있다[4-6]. 지하정보 변화객체 탐지추출 기술은 지하공간통합지도에 저장된 원본 지도와 새롭게 입력된 갱신 지도를 비교하여 변화가 발생한 객체를 선별하고 변화 객체를 신규 객체, 삭제 객체, 속성 변경 객체로 분류하여 지하공간통합지도를 갱신함으로써 갱신 시간을 대폭 단축시키는 역할을 하고 있다. 입력으로 사용되는 갱신 지도는 통관지도와 부분지도로 구분된다. 통관지도는 특정한 지방자치단체에서 관리하는 동일한 유형의 객체들을 모두 포함한 것을 말하며, 부분지도는 굴착작업을 통해 변경된 객체만 포함하는 지도이다. 부분지도를 이용한 변화 객체 탐지추출에서는 삭제 객체를 탐지하지 못하는 문제가 있다. 삭제 객체는 원본지도에는 존재하나, 갱신지도에는 없는 객체를 의미한다. 부분지도에는 신규로 매설된 객체(신규 객체)나 원본 객체를 대체한 객체(속성 변경 객체)만 포함되므로 삭제 객체는 존재하지 않는다. 그러나, 신규 객체가 원본 객체의 형상을 일부 변경하여 매설된 경우, 객체를 구성하는 포인트들에 대한 Exact Matching 방

법을 사용하는 현재의 변화 객체 탐지·추출 기술 [5,6]에서는 신규 객체와 원본 객체를 다른 객체로 판정하므로 신규 객체는 통합지도에 저장되고 원본 객체는 그대로 유지함으로써 동일 위치에 두 개 이상의 객체가 존재하는 문제가 발생한다.

본 논문에서는 부분지도에 포함된 갱신 객체 중 원본 객체의 형상과 일부가 같은 객체가 존재하는 경우, 갱신 객체와 중첩된 형상을 가지는 원본 객체를 검출한 후 이 객체들을 동일 객체로 간주하였다. 따라서, 검출된 원본 객체는 삭제 객체로 선정되어 지하공간통합지도에서 제거하고 갱신 객체는 통합지도에 저장함으로써 동일 위치에 하나의 객체만 존재할 수 있도록 지원하는 부분 변경 객체(이하, 본 논문에서는 부분 변경 객체, 중첩 객체, 동일 객체가 같은 의미로 사용) 탐지 및 추출 방법에 대하여 설명하였다. 중첩 객체 탐지 방법으로는 DE-9IM(Dimensionally Extended 9 Intersection Model), Relate Function, Intersection Function을 사용하였다. 지하시설물로 대표되는 선형 객체를 대상으로 시험한 결과 세 가지 방법 모두 오류 없이 중첩된 객체들을 동일 객체로 탐지하였으며, Relate function이 가장 빠르게 처리함을 확인하였다.

2. 관련 연구

동일 객체나 중첩 객체를 찾아내는 방법은 주로 중복 객체를 탐지하여 제거함으로써 데이터 세트의 신뢰성을 향상하기 위해 개발되었다. 본 논문에서는 속성을 제외한 지오메트리(Geometry)만을 사용하는 중복 객체 탐지 방법을 크게 세 가지로 구분하였다. 첫째는 객체들을 구성하는 모든 포인트가 일치할 때 동일 객체로 탐지하는 것으로 현재 변화 객체 탐지·추출 기술에서 사용 중이다. 둘째는 객체들 사이의 거리를 계산하여 일정한 거

리 이내의 객체를 동일 객체로 판정하는 방법이다 [7]. 셋째는 객체들 사이의 공간 관계를 분석하여 동일한 객체를 찾아내는 방법이다[8,9].

거리 계산 방법[7]은 객체 사이의 거리를 계산하는 방법과 객체들을 구성하는 포인트들 사이의 거리를 계산하는 방법으로 분류한다. 객체 거리 계산 방법은 두 객체를 구성하는 지오메트리 G_1 과 G_2 사이의 거리를 계산하여 이 거리가 주어진 한계값(T) 이하이면 동일 객체로 판정하는 것으로 $|G_1 - G_2| \leq T$ 를 만족하면 G_1 과 G_2 는 동일 객체이다. 포인트 거리 계산 방법은 두 객체를 구성하는 포인트들 사이의 거리를 계산하였을 때 모든 포인트 쌍의 거리가 한계값(T) 이하이면 동일 객체로 판정하는 것으로, 두 객체의 지오메트리 G_1 , G_2 가 각각 다음과 같은 포인트로 정의될 때, $G_1 = \{P_1, P_2, \dots, P_n\}$, $G_2 = \{Q_1, Q_2, \dots, Q_n\}$, 모든 포인트에 대해 $\forall i: |P_i - Q_i| \leq T, i=1, 2, \dots, n$ 을 만족하면 G_1 과 G_2 를 동일 객체로 판정한다.

관계 분석 방법은 DE-9IM[8], Relate 함수 및 Intersection 함수[9]로 구분할 수 있다. DE-9IM은 두 개의 지오메트리 사이의 공간 관계를 패턴 행렬 문자열을 이용해서 비교하는 방법으로 점 객체는 1차원, 선 객체는 2차원, 면 객체는 3차원으로 정의한다. 이 모델은 객체들 사이의 교차하는 차원을 고려하여 모든 유형의 지오메트리 사이에서 생성되는 공간 관계를 내부, 경계, 외부의 교차 쌍으로 분석하여 3 X 3 매트릭스를 생성한다. 그림 2는 두 다각형 사이의 패턴 행렬에 대한 예제이다.

패턴 행렬에 들어갈 수 있는 값들의 종류는 $\{T, F, *, 0, 1, 2\}$ 이다. 여기에서 T 는 교차가 존재해야 하며 교차 결과에 대한 차원 값인 $\dim$ 은 0, 1 또는 2가 될 수 있다. F 는 교차가 존재하지 않아야 하며, $\dim = -1$ 이어야 한다. $*$ 는 교차의 존재 여부와 관련 없으며, $\dim = -1, 0, 1$ 또는 2가 될 수 있다. 0은 교차가 존재해야 하며, $\dim = 0$ 이어야

한다. 1은 교차가 존재해야 하고, $\dim = 1$ 이어야 한다. 2는 교차가 존재해야 하고, $\dim = 2$ 이어야 한다. 따라서, 그림 2와 같이 면 객체들 사이에 중첩(overlap)이 발생하였을 때 패턴 행렬은 $\{T * T * * * T * *\}$ 로 표현된다.

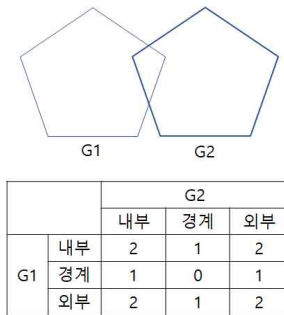


그림 2. 다각형 사이의 패턴 행렬 예제
Fig. 2. example of the pattern matrix of polygons

Relate(관계연산) 함수는 두 개의 지오메트리를 비교하여 이들에 대한 공간적 관계를 판정한다. 정의된 관계연산 함수는 Equals, Intersects, Disjoint, Crosses, Overlaps, Touches, Within, Contains와 같은 것들이 있으며, 각각의 함수는 True 또는 False를 반환한다.

Intersection 함수는 두 개의 지오메트리의 교차되는 부분을 표시하는 지오메트리를 반환한다. 즉, 교차되는 부분은 두 개의 지오메트리가 공유하는 부분이다. 지오메트리들 사이에 공유되는 부분이 없는 경우는 빈 지오메트리가 반환된다.

3. 부분 변경 객체 유형 분류

부분지도가 입력된 경우 삭제된 객체를 찾기 위해 2차원 공간에서 두 개 이상의 객체는 동일한 위치를 공유할 수 없음을 가정하였다. 즉, $G_1 \cap G_2 = \emptyset$ 이다. 이 가정에 따르면, 동일한 위치를

공유하는 두 객체가 존재하는 경우, 즉, $G_1 \cap G_2 \neq \emptyset$ 이면, 두 객체는 동일 객체로 판정하며, 부분적인 형상의 변화가 발생하거나 형상이 완전히 일치한 것으로 간주하였다. 따라서, 지하공간통합지도에 저장된 원본 객체는 삭제 객체, 부분지도에 포함된 갱신 객체는 신규 객체로 판정함으로써 삭제될 객체를 찾아내고 지하공간통합지도에서 제거하여 특정 위치를 공유하는 중복된 객체가 존재하지 않도록 하였다. 그러나, 두 개의 객체가 서로 교차하는 경우, 두 개의 객체가 한 점에서 만나는 경우는 각각 서로 다른 객체로 간주하여 원본 객체는 그대로 유지하고 갱신 객체는 신규 객체로 처리하여 지하공간통합지도에 저장하였다.

동일 객체로 판정되는 경우는 다음과 같이 두 가지로 구분하였다. 첫째, 원본 객체 및 갱신 객체의 길이가 연장, 감소 등의 변화가 발생한 경우로 원본 객체는 지하공간통합지도에서 삭제하고, 갱신 객체는 신규로 처리하여 지하공간통합지도에 저장한다. 이러한 경우는 그림 3에 표시하였다.

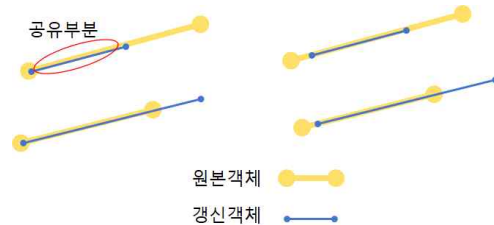


그림 3. 원본과 갱신 객체의 길이가 변경된 경우
Fig. 3. length of the original and updated objects changes

둘째, 원본 객체 및 갱신 객체의 형상 중 일부가 변화된 경우로 원본 객체는 지하공간통합지도에서 삭제하고, 갱신 객체는 신규로 처리하여 지하공간통합지도에 저장한다. 이러한 경우는 그림 4에 표시하였다.

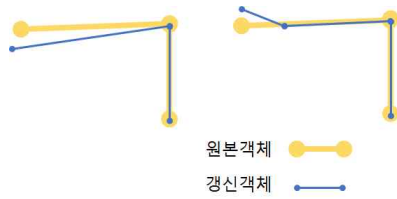


그림 4. 원본과 갱신 객체의 형상 중 일부가 변경된 경우
Fig. 4. some of the shapes of the original and updated objects have changed

또한, 원본 객체와 갱신 객체가 한 점에서 만나는 경우는 비록 두 객체가 서로 만나지만 다른 객체로 판정하여 원본 객체는 그대로 유지하고, 갱신 객체는 신규로 처리하여 지하공간통합지도에 저장한다. 이러한 경우는 그림 5에 표시하였다.

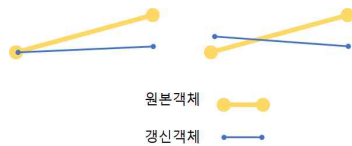


그림 5. 원본과 갱신 객체가 한 점에서 교차하는 경우
Fig. 5. the original and updated objects intersect at on point

4. 부분 변경 객체 검출 결과

4.1 개발환경

부분지도에 포함된 갱신 객체와 지하공간통합 지도에 포함된 원본 객체를 비교하여 부분 변경 객체인지를 판별하기 위해 DE-9IM, Relate 함수, 및 Intersection 함수를 적용하여 성능을 비교하였다. DE-9IM은 비교 대상 객체들의 관계에 대한 패턴 행렬을 반환하고, Relate 함수는 대상 객체들 사이에 공간 관계가 존재하는 경우 그 유형을 반환하며, Intersection 함수는 중첩되는 부분을 별도의 객체로 반환한다. 개발환경으로는 Windows 10을 사용하였으며, 개발 언어는 Java, 통합 개발

환경으로는 Eclipse를 사용하였다. 실험에 사용한 하드웨어 장비와 환경은 표 1에 표시하였다.

표 1. 개발환경
Table 1. development environment

CPU	11세대 Intel 코어 i7-1165G7
RAM	64G
SSD	4T
OS	Windows 10
Language	Java JDK 1.8
GPU	NVIDIA Geforce GTX 1650 4GB

4.2 실험용 데이터

일반적인 지하시설물 공간 데이터에서는 데이터의 중복이나 중첩을 기본적으로 허용하지 않으며, 이러한 데이터는 원칙적으로 오류로 검출되어 하나의 객체만 남기고 나머지는 삭제되어야 한다. 그러나, 작업자의 숙련도에 따라 일부 중복 또는 중첩 객체가 데이터에 포함되는 경향이 있다[10]. 따라서, 실제 데이터에서는 실험에 필요한 데이터의 존재 여부를 확인하고 추출하는 것은 시간이 오래 걸리는 작업이며 지하시설물은 보안 대상이라 데이터 획득 절차도 복잡하여 실험에 필요한 중복 및 중첩 데이터를 직접 생성하였다. 생성된 실험용 데이터는 그림 6에 표시하였다. 실험용 데이터는 원본 객체와 갱신 객체의 쌍으로 구성되어

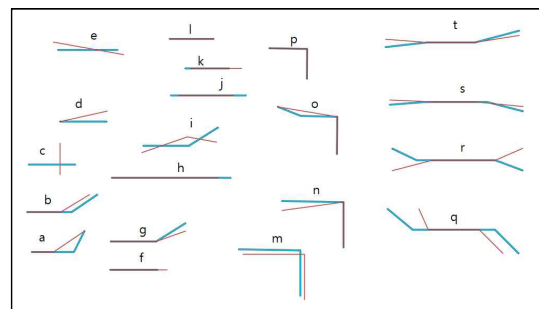


그림 6. 실험용 데이터 세트
Fig. 6. test data set

있으며, 원본 객체의 일부를 변경하여 갱신 객체를 생성하였다. 실험용 데이터는 두 개의 객체가 서로 교차하는 경우, 서로 중첩된 경우, 서로 형상은 일치하나 속성이 변경된 경우 등을 포함한다. 이 중 교차하는 경우는 서로 다른 객체로, 중첩된 경우는 동일 객체로, 형상은 같으나 속성이 다른 경우는 속성 변경 객체로 추출되어야 한다. 각 객체 쌍에 대한 특징은 표 2에 설명하였다.

표 2. 원본과 갱신 객체 쌍 특징
Table 2. characteristics of original and updated object pair

식별자	특징
a	원본 객체와 갱신 객체의 두 번째 포인트만 다름
b	원본 객체와 갱신 객체의 두 번째, 세 번째 포인트 상이
c	원본 객체와 갱신 객체가 크로스 (교차)
d	원본 객체와 갱신 객체의 시작 점 공유하고 형상 상이
e	원본 객체와 갱신 객체가 크로스 (교차)
f	갱신 객체의 길이가 연장됨, 시작 점 동일
g	갱신 객체의 두 번째 선분의 연결 각도 달라짐, 두 객체
h	갱신 객체의 길이가 감소, 시작 점 동일
i	원본 객체와 갱신 객체가 두 번 교차
j	갱신 객체가 원본 객체의 내부로 포함됨
k	갱신 객체의 시작 점이 이동하고 길이도 연장
l	원본 객체와 갱신 객체의 Geometry 동일, 속성 상이
m	원본 객체와 갱신 객체의 형상 동일, 위치 상이
n	갱신 객체의 두 번째 선분의 연결 각도 달라짐
o	갱신 객체의 포인트 하나 삭제 (원본 : 4개, 갱신: 3개)
p	원본 객체와 갱신 객체의 Geometry 동일, 속성 상이
q	원본 객체와 갱신 객체의 두 번째 선분 일정 부분 중첩
r	원본 객체와 갱신 객체의 두 번째 선분 일정 부분 중첩
s	원본 객체와 갱신 객체의 모든 포인트 상이, 두 번째 선분 일부 중첩
t	원본 객체와 갱신 객체의 두 번째, 세 번째 포인트 동일

4.3 처리 속도 비교

부분지도가 입력되었을 때 부분지도에 포함된 객체와 동일 객체를 선별하기 위한 DE-9IM, Relation, Intersection 방법의 처리 속도를 비교하였다. 비교 결과는 표 3에 표시하였다.

표 3. 객체 처리 시간 비교
Table 3. results of object processing time

객체 비교 방법	소요시간(S)
DE-9IM	1.405
Relate 함수	1.273
Intersection 함수	1.809

세 가지 방법 중 Relate 함수는 객체와 객체 사이의 공간 관계의 존재 여부를 touch, crosses, within, overlaps, contains이라는 다섯 가지로 구분하여 각각 true 또는 false 만을 판단함으로 가장 처리 시간이 빠르다. 만일, 형상이 완전히 동일한 객체인 경우는 within과 contains의 결과가 모두 true로 판정된다. DE-9IM 방법은 중복되는 부분에 대한 차원 값을 계산하여 패턴 행렬을 생성한 후에 이 패턴이 어떤 공간 관계 패턴에 해당하는지를 비교하는 시간이 추가되어 Relate 함수보다 처리 시간이 느리다. 반면, Intersection 함수는 객체들 사이에 중첩되는 부분을 추출하여 별도의 지오메트리로 생성하여 반환함으로써 처리 시간이 가장 오래 걸린다. 이렇게 추출된 객체는 다른 분석 작업에 활용될 수 있다.

4.4 부분 변경 객체 탐지 결과 비교

부분 변경된 객체들 사이에 동일 객체 탐지 성능을 비교하였다. DE-9IM, Relate 함수, Intersection 함수 모두 오류 없이 동일 객체를 100% 탐지하였다. 그림 6의 데이터에 개발 결과를 적용한 동일 객체 탐지 결과는 그림 7에 표시하였다. 동일 객체로 판정된 경우 원본 객체는 삭제 객

체로 분류되고, 갱신 객체는 신규 객체로 분류하였다. 또한, 형상이 동일하고 속성이 변경된 객체는 속성 변경 객체로 분류하였다. 기존의 방법에서는 원본 객체와 형상이 조금이라도 변경된 갱신 객체가 존재하는 경우, 원본 객체는 그대로 유지하고 갱신 객체는 신규 객체로 분류함으로써 동일 위치에 유사한 형상의 객체가 두 개 이상 존재하는 상황이 발생하였으나 제안된 방법을 적용하면 동일 위치에는 오직 하나의 객체만 존재한다.

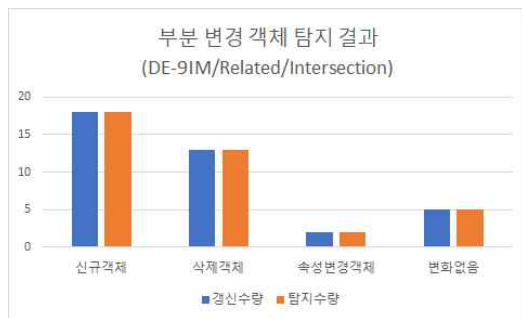


그림 7. 부분 변경 객체 탐지 결과
Fig. 7. results of extracting partially changed objects

그림 6에서 c, d, e, i, m 객체 쌍은 원본 객체와 갱신 객체가 교차하는 하지만 중첩되는 부분이 없으므로 서로 다른 객체로 분류하였다. 따라서, 원본 객체는 그대로 유지하고 갱신 객체는 신규 객체로 처리하여 저장된다. 또한, 그림 6에서 a, b, f, g, h, j, k, n, o, q, r, s, t 객체 쌍은 원본 객체와 갱신 객체들 사이의 중첩되는 부분이 존재하므로 원본 객체와 갱신 객체를 동일 객체로 판정하여 원본 객체는 삭제 객체로 분류하고 갱신 객체는 신규 객체로 처리하였다. 따라서, 신규 객체의 수량은 18개, 삭제 객체는 13개, 변화 없는 객체는 5개로 각각 판정되었다. 그리고, 그림 6에서 l과 p 객체 쌍은 원본 객체와 갱신 객체의 지오메트리가 동일하고 속성이 변경되었으므로 속성 변경 객체로 처리하였다.

그림 8은 객체 쌍 n에 대한 DE-9IM, Relate 함수, Intersection 함수의 처리 결과를 나타내고 있다. DE-9IM 방법은 비교 대상 객체들 사이의 공간 관계에 대한 패턴 행렬을 반환하고 이 값을 개별 공간 관계 패턴과 비교하여 두 객체 사이의 연관성을 판정한다. 실험에서 비교한 공간 관계는 Touch, Crosses, Within, Overlaps, Contains이며, 각각에 대한 패턴 행렬은 표 4에 표시하였다. 특히, Within과 Contains는 기준 객체 및 비교 객체의 선택에 따라 상반된 결과를 나타낸다. 본 실험에서는 원본 객체를 기준 객체, 갱신 객체를 비교 객체로 사용하였다.

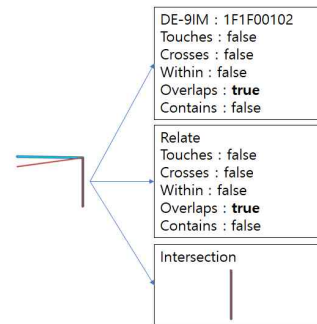


그림 8. 탐지 방법들의 판정 결과 비교
Fig. 8. comparison of results of three methods

표 4. DE-9IM 패턴 행렬
Table 4. DE-9IM pattern matrix

공간 관계	패턴 행렬
Touches	FT*****
Crosses	0*****
Within	T*F**F***
Overlaps	1*T***T**
Contains	T*****FF*

4.5 중첩 비율 비교

4.4에서 객체 비교는 조금이라도 중첩되는 부분이 있는 경우 동일 객체로 판정하였다. 이 경우

중첩률이 10%인 객체들과 90%인 객체들이 모두 동일 객체로 판정된다. 그러나, 중첩 비율이 10%인 객체들을 동일 객체로 판정하기에는 무리가 있다는 생각이 들 수 있다. 따라서, 이번 실험에서는 객체들 사이의 중첩률을 비교하였다. 비교 결과는 그림 9에 표시하였다.



그림 9. 원본과 갱신 객체 사이의 중첩 비율
Fig. 9. comparison of overlapping ratio between original and updated objects

그림 9에서 보는 것처럼, 중첩 비율이 0%인 c, d, e, m, i 객체 쌍은 서로 다른 객체로 판정되었고, 중첩 비율이 1인 l과 p 객체 쌍은 동일 객체이나 속성이 변경된 객체로 판정되었다. 이 객체 쌍들을 제외하고 나머지 객체 쌍의 중첩 비율은 [0.3, 0.89] 사이에 존재한다. 만일 80% 이상 중첩되어야 동일 객체로 선정하면 2개의 객체만이 동일 객체로 탐지된다. 이 비율을 50%로 낮추면 5개의 객체가 동일 객체로 탐지된다. 즉, 중첩률에 따라 동일 객체로 판정되는 객체들에 변경이 발생한다. 이렇게 특정 기준에 따라 결과가 변경되는 경우 분류기준을 정하는 것이 가장 중요하다. 따라서, 중첩 비율을 사용하려고 하는 경우 최적의 중첩률을 찾아내는 연구가 필요하며, 누구나 동일 객체 판정 결과에 동의할 수 있는 최적 중첩률은 수많은 실제 데이터를 확인한 후 결정할 수 있을 것으로 생각된다.

4. 결 론

본 논문에서는 변화된 객체만을 선별하여 지하 공간통합지도에 반영함으로써 통합지도 갱신 시간을 단축하는 과정에서 부분지도가 입력된 경우 삭제 객체를 검출하지 못하는 문제를 해결하는 방법을 제안하고 성능을 평가하였다. 삭제 객체를 검출하기 위해서는 비교 대상이 되는 두 개의 객체가 동일 객체 또는 중첩 객체인지를 먼저 판정해야 하며, 2차원 평면상에 존재하는 객체들이 같은 영역을 공유하는 경우 동일 객체로 판정하였다. 이러한 규칙은 길이가 변화된 객체, 전체 형상 중 일부분이 변경된 객체 등을 효과적으로 탐지할 수 있다. 객체들 사이에 공유되는 부분의 존재 여부를 판정하기 위해 DE-9IM, Relate 함수, Intersection 함수를 적용하였다. 세 가지 방법 모두 비교 대상이 되는 두 가지 객체들의 공간에서의 연관관계를 평가하는 공통점이 있다. 세 가지 방법을 선형으로 구성된 지하시설물 데이터에 적용하였을 때, 모든 변화가 발생한 객체들을 오류 없이 탐지하였으며, 처리 속도는 Relate 함수가 가장 빠르다는 것을 확인하였다. 따라서, 동일 객체 판정을 위해서는 Related 함수를 사용하는 것이 바람직하다. 논의된 방법들은 삭제 대상 객체를 탐지함으로써 동일 위치에 두 개 이상의 중복 객체가 지하공간통합지도에 저장되는 상황을 방지하는 역할을 한다. 그러나, 논의된 방법들은 중첩률에 따라 동일 객체로 판정되는 경우에 변화가 발생한다. 이러한 변화로 인하여 삭제된 객체가 부분지도에 포함되어 입력되기 보다는 별도의 파일로 분리되어 입력되는 것이 바람직하다. 따라서, 입력되는 파일 종류의 일관성 유지를 위해 하나의 파일로 입력되었을 때 비교를 통해 동일 객체로 판정할 수 있는 기준이 되는 최적의 중첩률을 선정하는 추가적인 연구가 필요하다.

본 연구는 국토교통부/국토교통과학기술진흥원의 지원으로 수행되었음 (과제번호 RS-2020-KA158151).

참 고 문 헌

- [1] Jiha Safety Information System. (2023). <https://ww.jis.go.kr>
- [2] Kiyong Lee, Sangjun Kang. (2014). "Sinkhole Swallowing the City, Causes and Solutions", Report of Gyeonggi Research Institute, <https://www.gri.re.kr/web/contents/webSearch.do?kwd=sinkhole>
- [3] Dong Hyun Park, Yong Gu Jang, Ji Song, Ryu. (2021). "Design of Standard Submission Format for Underground Structures : An Automated Update of the UnderSpace Integrated Map", Journal of the Korean Society of Surveying, Geodesy, Photogrammetry and Cartography, Vol.39, No. 6, 469-476, DOI: <https://doi.org/10.7848/ksgpc.2021.39.6.469>
- [4] Chan-min Jeon. (2022). "Ensure up-to-dateness, accuracy, and usability of integrated underground space maps for underground safety management.", Engineering Journal, <http://www.engjournal.co.kr/news/articleView.html?idxno=1937>
- [5] Kwangsoo Kim, Heyung-Sub Lee, Juwan Kim. (2020). "A Study on Updated Object Detection and Extraction of Underground Information", Journal of Software Assessment and Valuation, 16(2), pp. 99-108, DOI: <http://dx.doi.org/10.29056/jsav.2020.12.11>
- [6] Kwangsoo Kim, Bong Wan Kim, Insung Jang. (2021). "Detecting and Extracting Changed Objects in Ground Information", Journal of the Korean Society of Surveying, Geodesy, Photogrammetry and Cartography, Vol.39, No.6, pp.515-523, DOI: <https://doi.org/10.7848/ksgpc.2021.39.6.515>
- [7] ESRI, Find Identical(Data Management), <https://desktop.arcgis.com/en/arcmap/latest/t/>

<tools/data-management-toolbox/find-identical.htm>

- [8] Open GIS Consortium. (2011). OpenGIS Implementation Standard for Geographic Information - Simple Features Access - Part1: Common Architecture, version 1.2.1, <https://www.ogc.org/standard/sfa/>
- [9] Mark Leslie, Paul Ramsey, et. al. (2022) Introduction to PostGIS, <https://postgis.net/workshops/postgis-intro/>
- [10] Kwangsoo Kim, Insung Jang, Bong Wan Kim. (2023). "A Detection Method of Duplicated Objects in Underground Facility Maps", In Proceedings of Spring Conference of Korea Spatial Information Society, pp.79-80, <https://www.dbpia.co.kr/journal/articleDetail?nodeId=NODE11428165>

저 자 소 개



김광수(Kwangsoo Kim)

1993.2 고려대학교 정보공학과 졸업
 1995.2 고려대학교 전산학과 석사
 2016.8 충남대학교 컴퓨터공학과 박사
 1994.12-현재:한국전자통신연구원 책임연구원
 <주관심분야> 공간정보, LBS, 사물인터넷



김봉완(Bong Wan Kim)

1992.2 한양대학교 전자공학과 졸업
 1994.2 KAIST 전기및전자공학과 석사
 2000.8 KAIST 전자전산학과 박사
 2019.7-2019.12 Univ. of Strathclyde 방문연구원
 2000.10-현재:한국전자통신연구원 책임연구원
 <주관심분야> 사물인터넷, 데이터분석, 안전관제