Ph.D. DISSERTATION

Generation of an Indoor Graph Database for People with Mobility Disabilities from Scanned Floor Plans

스캔 도면을 활용한 이동약자용 실내 그래프 데이터베이스 구축

August 2021

Seoul National University
Department of Civil and Environmental Engineering
Seula Park

Generation of an Indoor Graph Database for People with Mobility Disabilities from Scanned Floor Plans

스캔 도면을 활용한
이동약자용 실내 그래프 데이터베이스 구축

지도교수 유 기 윤

이 논문을 공학박사 학위논문으로 제출함
2021년 4월

서울대학교 대학원
건설환경공학부
박 슬 아

박슬아의 박사 학위논문을 인준함
2021년 6월

위 원 장 ____________________

부위원장 ____________________

위 원 ____________________

위 원 ____________________

위 원 ____________________

Abstract

Changes to the indoor environment have increased social interest in ensuring the mobility of people with disabilities. Therefore, the demand for customized indoor routing services for people with mobility disabilities (PWMD), who have many travel restrictions, is increasing. These services have progressed from spatial routing to personalized routing, which reflects personal preferences and experiences in planning an optimal path. In this regard, it is necessary to generate a database for PWMD with a flexible schema suitable for the efficient manipulation and processing of data.

This study aims to propose a technique of generating an indoor graph database for PWMD using scanned floor plans. First, a conceptual data model was developed by deriving relevant indoor features and influential factors, considering various international regulations on indoor environments. Also, the accessibility index was designed based on the data model to quantify the difficulties in accessing spaces based on each indoor space's geometric characteristics. Next, a three-stage process was proposed: retrieving the structure of spaces from scanned floor plans through a transfer learning-based approach, retrieving topology and assessing accessibility for creating an indoor network model for PWMD, and converting the network model into a graph database. Specifically, an indoor structure map is created by fine-tuning the modified ResNet-based model with newly annotated floor plans for extracting structure information. Also, based on the spatial relationship of the extracted features, the indoor network model was created by abstracting indoor spaces with nodes and links. The accessibility of each space is determined by the proposed indices and thresholds; thereby, a feasible network for PWMD could be derived. Then, a process was developed for automatically converting an indoor network model, including accessibility property, into a graph database.

The proposed technique was applied to the Seoul National University dataset to generate an indoor graph database for PWMD. Two scenario-based routing tests were conducted using the generated database to verify the utility of results: multi-floor routing and integrated indoor-outdoor routing. As a result, compared with the path for general pedestrians, the optimal path for PWMD was derived by avoiding inaccessible spaces, including vertical movement using elevators rather than the nearest stairs. In other words, applying the proposed technique, a database that adequately described an indoor environment in terms of PWMD with sufficient mobile constraint information could be constructed. Moreover, an integrated indoor-outdoor routing could be conducted by only creating an entrance-labeled relationship, without scale and coordinate transformation. This result reflects the usability of the generated graph database and its suitability regarding the incorporation of multiple individual data sources.

The main contribution lies in the development of the process for generating an indoor graph database for PWMD using scanned floor plans. In particular, the database for PWMD routing can be generated based on the proposed data model with PWMD-related features and factors. Also, sub-procedures for topology retrieval and graph database conversion are developed to generate the indoor graph database by the end-to-end process. The developed sub-procedures are performed automatically, thereby reducing the required times and costs. It is expected that the target database of the proposed process can be generated considering utilization for various types of routing since the graph database is easily integrated with multiple types of information while covering the existing spatial model's function.

Keywords : graph database, transfer learning, network model, data model, people with mobility disabilities, accessibility

Student Number : 2017-31313

Contents

List of tables

List of figures

Terminology

■ Cypher

: A declarative graph query language for efficient querying in a property graph

■ Data model

: An abstract model organizing data elements with their relations and standardizing how they relate to the properties of actual entities

■ General Space

: An navigable space for activities such as rooms, terraces, and lobbies

■ Graph database

: A database based on graph models, composed of nodes, edges, and properties to represent data, for semantic queries

■ Indoor network model

: The model which describes the indoor environment both geometrically and topologically

■ PIT

: Passage for electric wiring, water pipes, and gas pipes

■ Transition Space

: A space providing passage between indoor spaces such as corridor, stair

■ Unified Modeling Language (UML)

: A modeling language used in the software engineering field providing a standard way to visualize the design of a system

1. Introduction

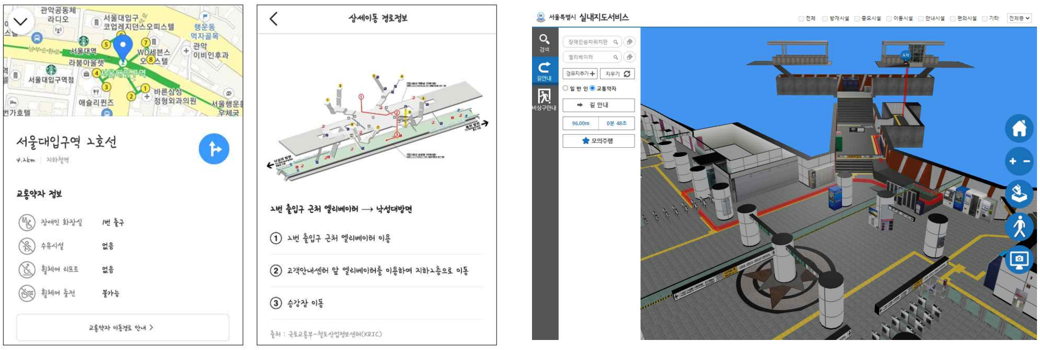
1.1 Objectives and contributions

As people spend more time indoors and their activities diversify, indoor spaces are gradually increasing in size and becoming more structurally complex. The mobility of people with disabilities is the most challenging and important social issue regarding these indoor environmental changes.

Unlike outdoor spaces, indoor spaces are modeled with objects such as walls and doors. Indoor spaces are recognized as constrained spaces that are restricted by such objects (Li & Lee, 2013). This characteristic of indoor spaces may cause great difficulty for people with mobility disabilities (PWMD) when attempting to freely pass through indoor spaces. Therefore, navigation services, including information on travel restrictions, must be provided. However, the practical application of existing indoor data to those services is limited by the absence of models with reasonable representations of indoor environments for a specific purpose (Isikdag *et al.*, 2013). Therefore, to develop useful services, an indoor database needs to be generated with sufficient information related to the mobility of PWMD. The target data must be composed of features related to PWMD's movements and must include information on the significant factors affecting their travel. It is also necessary to generate an indoor database based on a standardized data model to increase the usability of the data.

Despite the increasing demand for services for PWMD, limited information for PWMD is provided comparing with navigation for general pedestrians. Kakao, which operates web and mobile map services, launched an application-based route guidance service for PWMD in July 2020 ([Figure 1-1(a)]). This service provides information on transfer routes and convenience facilities in 1,107 subway stations nationwide. However, Kakao map's service only provides accessible route information for each exit in the form of images and texts; it does not provide interactive navigation. Meanwhile, Seoul has been operating an indoor map service[1], which began with a pilot service in 2015. This includes a route guidance service for the PWMD ([Figure 1-1(b)]). Since 2013, people have been employed to construct two-dimensional (2D) and three-dimensional (3D) data based on floor plans and field surveys for subway stations and major public facilities. Large amounts of costs and human resources are being invested into building data, with a high manual work rate. In other words, Seoul's service enables interactive navigation and provides a high level of information based on 3D visualization, but high costs are required to generate the database. Also, Kakao map is intended to provide information only for subway stations, and Seoul's indoor map is limited to subway stations and public facilities. Consequently, it is essential to generate a database for PWMD, including constraint information for providing valuable services. Simultaneously, an efficient method of generating a database needs to be developed since PWMD-related information should be updated and managed sustainedly. In detail, efficient database generation can be realized by using inexpensive source data or developing automatic construction techniques. If the database can be generated more economically and rapidly, then these service areas can likely be expanded, and the quality of services using the database will be improved.

1) http://indoormap.seoul.go.kr/



(a) Kakao map application (b) Indoor map service of Seoul

[Figure 1-1] Route guidance services for PWMD

Recently, routing types have diversified to consider preferences while reflecting travel constraints. For example, a user tend to select an optimal path considering preferences such as high familiarity, low congestion, and fewer intersections and turns (Pang & Chu 2007; Peeta & Yu, 2005). In particular, PWMD, who have special needs when moving, not only need to find the optimal path based on travel distance or time but also require guidance that reflects user preference.

Based on previous spatial routing, routing services are extended to cover semantic routing, using a well-defined representation of location knowledge as indoor semantics. Specifically, a more helpful route to the PWMD can be provided through semantic routing, which is based on the various indoor semantic information. Functional categories of space such as horizontal/vertical unit and connector are one of indoor semantics (Liu *et al.*, 2019). Using this semantic, for example, a path with hardly inter-floor transition can be guided for PWMD by minimizing the number of vertical units in the route. Moreover, the emergence of semantic routing enables hierarchical (coarse-to-fine) routing through incorporation with spatial routing. Furthermore, it is necessary to guide a route reflecting the context of the indoor environment since PWMD is highly affected by many indoor environmental constraints. For example, Routes that avoid temporarily restricted sections or overcrowded areas should be provided for PWMD. Context-aware routing creates friendly routes in consideration of real-time user context information with environmental context, and personalized routing can reflect users' experiences and preferences (Alqahtani *et al.*, 2018; Bian *et al.*, 2014; Cheraghi *et al.*, 2019; Costa & Chrysanthis, 2019; Al Delail *et al.*, 2012; Guo *et al.*, 2020; Liu *et al.*, 2019). To provide a customized service for the PWMD, the service should be extended gradually to a routing that reflects the preferences and experiences of the PWMD while providing detailed information on the indoor environment.

For diversified and customized routing, it is required to generate a base database in consideration of the service expansion. The current spatial routing services utilize geometric and topological models. However, a structured indoor database such as a network model is inappropriate regarding the concurrent usage of unstructured data such as textual information. An alternative process of combining personalized information with indoor space data is currently emerging (Wu *et al.*, 2018).

In conclusion, for the various routing types, it is necessary to generate a database with a flexible schema that allows the efficient manipulation and processing of data. Existing relational databases are formally organized with tables of data items; they depend on rigid schema. Therefore, they need complicated processes to add new objects and their relations. The graph database has been developed to overcome all of the boundaries of relational databases (Jaiswal & Agrawal, 2013); thus, a graph database can be a practicable alternative.

Graph database represents data by graph models, composed of nodes, edges, and properties, for semantic queries. This type of database uses a simple property-graph model for efficient data traversals. Also, the graph database is optimized for processing interrelated datasets efficiently (Hor *et al.*, 2018) and avoids complex join operations (Steinmetz *et al.*, 2018). Furthermore, a graph database with a flexible schema is helpful for manipulating irregular data with a high level of performance, ensuring the scalability of data (Fernandes & Bernardino, 2018). A graph database is advantageous in integrating and utilizing structured and unstructured data because of its flexible schema. Also, it is convenient to add, delete, and change data due to little physical constraints; thus, the database can be updated rapidly. Therefore, the graph database is suitable as a database of PWMD navigation services that require various information and efficient maintenance.

With this background, a process for generating an indoor graph database for PWMD navigation, using scanned floor plans is proposed in this study. The resultant indoor graph database is generated upon a data model composed of PWMD-related features that are extracted from several indoor space standards. Moreover, most sub-procedures that comprise the entire process, from scanned floor plans to the indoor graph database, are automated. This will reduce the time and costs required to establish an initial indoor database for PWMD navigation.

The main contribution of this study is to develop the entire process for generating the indoor graph database for PWMD by retrieving the relevant information from the scanned floor plans. Previously, the data model which reasonably represents indoor environments in terms of PWMD has been absent. Moreover, the existing techniques for generating a network model or graph database have used semantically rich data sources, thus not being applicable to scanned floor plans. In this study, a conceptual data model configured with PWMD-related features and factors is developed to generate an appropriate database for PWMD routing services through the proposed process. Also, to generate a target database with an end-to-end process, automatic sub-procedures are developed for topology retrieval and graph database conversion that existing methods cannot cover. That is, the proposed process enables generating the indoor graph database for PWMD using scanned floor plans.

The target database is based on the data model designed with PWMD-related indoor features by referring to local and international regulations on indoor environments. Therefore, important information related to the accessibility of PWMD can be generated and stored in the database. The database is generated based on the data model, which is in a standardized form, thus being easily shared. Furthermore, the time and costs required to generate the indoor database for PWMD can be reduced as sub-procedures are conducted automatically for generating a network model and a graph database. Moreover, the proposed process enables generating a database economically since scanned floor plans, easily acquired without extra cost, are used as source data.

The database generated through the proposed process is expected to be used as a fundamental database for the diversified routing services for PWMD. The graph database is easy to maintain with a flexible schema and has few restrictions on the data format to be stored. Due to these characteristics of the graph database, the target graph database can even link unstructured information (e.g., a textual description of temporary obstacles, time information of accessible entrances, or reviews of places reflecting user experiences) to the existing spatial model, which is a relational database such as a network model. Also, as the graph database and the current network model are commonly based on a graph structure, the proposed indoor graph database, while covering the network model's function, can easily be combined with other types of data.

1.2 Related works

Indoor databases for PWMD navigation largely involve three issues. The first is on the way the indoor environment is conceptualized for specific purposes. In consideration of utilization in PWMD navigation, indoor environments should be described differently, and a data model composed of PWMD-related features needs to be designed. The second is to create an indoor spatial model from various data sources adequately. Also, there is an issue related to the technique that reduces the construction cost while correctly describing the indoor environments. For the path-finding for PWMD, the feasibility and accessibility of routes are more important than the time and distance costs (Evcil, 2012). An indoor spatial model with accessibility semantics is a fundamental element for indoor path planning by users with specific preferences (Ge *et al.*, 2015). The third issue accordingly is on relevant accessibility information for PWMD. This section presents a thorough review of studies related to these three issues.

1.2.1 Indoor environment conceptualization

When generating an indoor database suitable for various spatial services such as navigation, a conceptual model should be designed that reflects the environmental characteristics regarding service purposes. Several Previous studies have conceptualized the objects constituting indoor environments with relationships between them, and have suggested a schema for specific applications (Iida *et al.*, 2015; Isikdag *et al.*, 2013; Jung & Lee, 2017; Kang & Li, 2017; Kim *et al.*, 2014; Li *et al.*, 2018; Liu & Zlatanova, 2012; Maheshwari *et al.*, 2019; Srivastavaa *et al.*, 2018).

Most studies covering indoor space modeling are based on the indoor space standard, IndoorGML. OGC® IndoorGML standard (OGC IndoorGML, 2020) defines an open data model of indoor spatial information. The IndoorGML mainly focuses on modeling indoor spaces in terms of navigation purpose, thus specifying the standard for the exchange of geoinformation required for indoor navigation systems. IndoorGML, an application schema, comprises two major modules: a core module and an extension module. The core module defines the general concept of indoor spaces and expresses geometric and topological characteristics of indoor environments. The indoor navigation module, representative of the extension module, represents spaces by focusing on indoor travel. The navigation module involves a navigable space feature while including all core module contents. Kang & Li (2017) investigated the basic concept of IndoorGML as a cellular space model and discussed implementation issues for multiple purposes. Kim *et al.* (2014) developed the Indoor Spatial Data Model (ISDM), which is the integration of the IndoorGML topology model and the feature model in the CityGML Application Domain Extension (ADE). This data model supports the construction of data for several spatial services within indoor environments. Jung & Lee (2017) proposed an IndoorGML-based data model to support indoor patrol services. Their presented data model connects omnidirectional indoor images and indoor topological data based on the concept of IndoorGML. Srivastavaa *et al.* (2018) highlighted that IndoorGML has a structural problem in that it is limited to storing semantic information; thus they proposed a semantic extension to IndoorGML. Similarly, Maheshwari *et al.* (2019) presented a data model for extending IndoorGML, focusing on semantic information. They proposed a space classification model considering both the semantic and geometric characteristics of the space; the classes of the proposed model include syntactic and semantic components.

Several individual schemas, which is irrelevant to IndoorGML, have also been designed for specific applications. Liu & Zlatanova (2012) proposed an Indoor Navigation Space Model (INSM) that included an extended categorization of indoor spaces based on semantics. Isikdag *et al.* (2013) designed a Building Information Modeling (BIM) Oriented Indoor Data Model (BO-IDM) for indoor navigation. They also presented a method for converting standard BIM/Industry Foundation Classes (IFC) data into the proposed BO-IDM model.

The lack of appropriate and applicable geometry and semantics limits the usability of the model. Therefore, a model for a service that targets users with particular purposes should include detailed knowledge of the environment (Isikdag *et al.*, 2013). Several researches related to the indoor model considering people with special needs, including PWMD, have been conducted in this context. The INSM (Liu & Zlatanova, 2012) includes the obstacle feature, which refers to a space that pedestrians cannot access. This definition is too broad to be used for PWMD navigation; thus, more sophisticated modeling is required. Iida *et al.* (2015) presented the concept of a standardized data model with landmarks. Their model comprised an IndoorGML extension for a voice navigation system, so the mobility constraints have not been sufficiently reviewed. Li *et al.* (2018) investigated an indoor space dimensional model designed for supporting barrier-free path-finding. They used a concept of a hybrid model mixed with geometric, topological, and semantic layers. Although the accessibility of wheelchair users was reflected in this model, the information was limited to door accessibility.

It is necessary to share indoor models and data sets using standardized data models to ensure the efficient development of services targeting users with special needs and preferences (Iida *et al.*, 2015). However, existing researches for modeling indoor spaces and schemas have not covered detailed descriptions specifically for PWMD indoor navigation. Furthermore, these studies have excluded detailed information in terms of mobile constraints. In conclusion, a much higher level of feature modeling needs to be conducted that reflects the complex properties of indoor elements while considering the actual application based on IndoorGML, which has good expandability. [Table 1-1] summarizes major related studies of indoor environment conceptualization.

[Table 1-1] Summary of major related studies
of indoor environment conceptualization

Previous studies	Base standard	Key issue
Kim *et al.*(2014)	CityGML, IndoorGML	• Integration of topology and feature model
Jung & Lee(2017)	IndoorGML	• Connection of omnidirectional indoor images and topological data • For indoor patrol services
Iida *et al.*(2015)		• Use of landmarks • For visually impaired • For voice navigation
Srivastavaa *et al.*(2018), Maheshwari *et al.*(2019)		• Difference in output database type
Isikdag *et al.*(2013)	BIM	• Use of semantic information
Liu & Zlatanova(2012)	-	

1.2.2 Indoor data construction

Indoor routing is to plan the optimal path using an indoor navigation model, which is represented by networks (Liu *et al.*, 2019). These navigation networks are configured with nodes and links representing the abstraction of spaces and their relationships. In particular, in a large complex building, information for finding the desired destination can be derived through network-based routing (Dao & Thill, 2018). Therefore, generating network models well-describing indoor environments is a critical issue for valuable navigation services. With this background, various studies have been conducted related to the creation of indoor network models. Those studies have been developed according to the differences between the source data and the construction technique, or the differences in the purposes of the models.

Many researchers have proposed methods to create indoor network models that elaborately describe the geometric and topological characteristics of indoor environments. Yang & Worboys (2015) presented a comprehensive approach to creating navigation graphs representing indoor environments. Their proposed process was almost automated; it uses the building plans as source data, including the geometric structure of indoor environments. A structure map that expresses each space's boundary as vertices and edges is then created from floor plan data using Computer-Aided Design (CAD). Next, a dual map that connects adjacent areas and a skeleton graph that expresses the connection of spaces through doors are generated. The topology retrieval of the final navigation graph is based on skeleton graph. Similarly, Srivastavaa *et al.* (2018) used architectural plans in CAD drawings as input data. They presented a workflow that organized the derivation of semantic information, based on topological relationships extracted from relevant geometric features in CAD drawings. Though both approaches were introduced as extensible methods to multi-floor buildings, they mainly focused on a single floor. Clementini & Pagliaro (2020) proposed a generation method that assumes the input spaces to be enclosed polygons though omitting the exact condition for input data. Furthermore, Lewandowicz *et al.* (2019) adopted the base points on the walls constituting each space as source information. Kim *et al.* (2015) used crowdsourced trajectory data to derive indoor network information. Mortari *et al.* (2019) and Wang & Niu (2018) started building a network from a floor plan drawn with Java OpenStreetMap editor (JOSM); this networks' inputs were similar characteristics to CAD drawings.

The majority of previous studies have developed techniques to create indoor network models from BIM (Fu *et al.*, 2020; Hamieh *et al.*, 2020; Khalili & Chua, 2015; Taneja *et al.*, 2011; Taneja *et al.*, 2016; Yan *et al.*, 2020). Researchers have proposed a methodology that focuses on determining how to conduct the abstraction of each space. For example, when modeling a corridor space, Taneja *et al.* (2011) applied Medial Axis Transformation (MAT). Moreover, Taneja *et al.* (2016) extracted the centerline through Straight-MAT (S-MAT), which is a modification of MAT. Fu *et al.* (2020) also adopted a straight skeleton-based abstraction.

Meanwhile, several researches have focused the detail level of the network model. Hamieh *et al.* (2020) proposed a methodology for creating an indoor network model that expresses three different levels of information: a macro graph involving simple connectivity, an external graph considering transition, and an internal graph representing the insides of individual spaces. Yan *et al.* (2020) differentiated the target space and derived a network by focusing on semi-indoor spaces that were not completely enclosed, even if the upper part of the space was blocked. Although there are differences in the issues that have been addressed, these studies all suggest automatic or semi-automatic techniques for creating a reasonable network model. Moreover, they use similar data sources to derive networks.

Furthermore, some studies have designed indoor network models for specific purposes, such as evacuation (Song *et al.*, 2021; Mirahadi & McCabe, 2021) and indoor and outdoor combined routing (Teo & Cho, 2016). However, these studies have followed the approach of supplementing information after the geometric network model has been established. In this case, the model was created by slightly modifying the techniques of the previous studies.

With the development of deep learning technologies, various data sources are being used. Typically, as scanned floor plans also exist for buildings built before the adoption of CAD or BIM, a vast amount of this type of data has been accumulated. Furthermore, scanned floor plans accurately describe the structures of spaces. However, since semantic information is lost during the scanning process, the key for constructing an indoor data with scanned floor plans is to retrieve said information. In this context, many studies have attempted to reconstruct indoor data using floor plan images (Jang *et al.*, 2020; Kim *et al.*, 2021; Liu *et al.*, 2017; Zeng *et al.*, 2019). These studies aimed to reconstruct indoor spatial model by segmenting walls and detecting objects such as doors, stairs, and elevators using various deep learning-based approaches. Their constructed data are in the form of 2D or 3D vectors that can be directly used for indoor maps.

Few studies have focused on reconstructing topology information using scanned floor plans, and they retrieved a low level of topology, which is not suitable for routing. Yamasaki *et al.* (2018) focused on retrieving topology by detecting geometric elements in floor plan images. They achieved this by using fully convolutional networks and classifying each room's usage. Niu & Song (2019) attempted to extract indoor navigation graphs from the building design through the faster Region Based Convolutional Neural Network (R-CNN) approach. However, the resulting former graphs were connectivity-based topological models and had a low level of geometric detail. The latter graphs, meanwhile, also showed the adjacency of rooms, but did not provide enough information to be used for actual navigation. Kim *et al.* (2021) detected walls and other objects (doors and stairs) through morphology conversion and the deep learning technique, respectively. They then created a topology map using detected objects. However, only connecting each node with the closest nodes is performed, and the sensor data were used as auxiliary data.

In summary, although scanned floor plans are valuable data sources for creating indoor network models, the techniques for generating indoor network models discussed above use semantically rich data sources, such as BIM in IFC or building plans in a vector format. Scanned floor plans suffer from more information loss than data sources such as BIM; thus, the techniques proposed in previous studies cannot be applied directly. Information of indoor structure and relations in scanned floor plans must be retrieved in advance. Some studies have used scanned floor plans, but these studies encountered limitations regarding creating practical network models. Therefore, it is essential to develop an efficient approach for creating indoor network models from scanned floor plans.

Interest in the diversification of indoor routing has increased following the development of construction techniques and the utilization of various data sources. However, there are two significant issues regarding routing diversification. The first issue is the appearance of coarse-to-fine routing (Gu *et al.*, 2019; Liu, 2017; Liu, & Zlatanova, 2013; Teo & Cho, 2016) related to the level of information for routing. Liu & Zlatanova (2013) proposed a two-level strategy for roughly determining a space sequence using only the connectivity information between spaces. It then searches for a minute path based on a geometric network, according to the selected space sequence, as the next step. The second issue concerns extending routing using various types of semantics, context, and personalized information (Afyouni *et al.*, 2013; Jabbar & Bulbul, 2019; Winter *et al.* al., 2018; Wu *et al.*, 2018).

As routing types have diversified, the existing databases for services such as the indoor network model need to be also expanded to include various types and structures. In terms of these issues, a graph database can be a practical alternative. Graph database technology comprises a compelling set of tools for modeling, storing, manipulating, querying, and managing graphs. It focuses on the attributes and relationships between entities in the data model (Hor *et al.*, 2018). The coarse-routing can be performed based on the relationships stored in the graph database. Simultaneously, detailed path information can be provided by combining the properties and values of each node and relationship. Furthermore, this database can be used in association with not only the relational database but also with various unstructured data, due to it having fewer structural restrictions.

Some studies have attempted to express an indoor spatial model using a graph database in this regard (Hor *et al.*, 2018; Ismail *et al.*, 2017; Ismail *et al.*, 2018; Xu, 2018). Hor *et al.* (2018) designed an architecture for implementing the BIM-GIS integrated Resource Description Framework (RDF) graph database. Ismail *et al.* (2017, 2018) presented a workflow for automatically transforming IFC models into labeled property graph models. The indoor spatial model, transformed as a Neo4j graph database[2], has also used in a case study for emergency routing. Moreover, graph database has been applied to evaluate the similarity between floor layouts (Sabri *et al.*, 2017). A review of related studies reveals that research into constructing indoor spatial models using graph databases has been concentrated on BIM in IFC, which contains predefined relationships between objects. To effectively use various routing types, a graph database needs to be generated using multiple sources, including existing information such as the indoor network model.

Consequently, it is necessary to generate a indoor database by expanding the data sources, which previously were focused on BIM/IFC. Scanned floor plans are useful sources since they are easy to acquire and contain various relevant information. Concurrently, the target database needs to have high usability and flexibility. Furthermore, an technique for quickly, economically, and efficiently generating databases should be developed. [Table 1-2] provides a summary of major related studies reviewed above.

2) An open-source, NoSQL, native graph database that uses pointers to navigate the graph

[Table 1-2] Summary of major related studies of indoor data construction

Previous studies	Source data	Target	Key issue of approach
Yang & Worboys (2015); Srivastavaa *et al.* (2018)	CAD	Network model	• Difference in modeling algorithm of network
Taneja *et al.* (2011), Taneja *et al.* (2016), Fu *et al.* (2020)	BIM/IFC		
Khalili & Chua (2015)			• Difference in schema
Hamieh *et al.* (2020)			• Improvement in level of details of output
Yan *et al.* (2020)			• Difference in target space for network derivation
Teo & Cho (2016), Mirahadi & McCabe (2021), Song *et al.* (2021)			• Difference in purpose of output network
Lewandowicz *et al.* (2019), Clementini & Pagliaro (2020)	Basic vector geometries		• Difference in modeling algorithm of network
Mortari *et al.* (2019), Wang & Niu (2018)	JOSM floor plan		

Previous studies	Source data	Target	Key issue of approach
Liu *et al.*,(2017), Zeng *et al.*,(2019), Jang *et al.*,(2020), Kim *et al.*,(2021)	Floor plan images	Geometry model	• Application of deep learning approach
Yamasaki *et al.*(2018), Niu & Song(2019)		Topological model	
Kim *et al.*(2021)	Floor plan images with auxiliary sensor data	Network model	
Ismail *et al.*(2017), Hor *et al.*(2018), Ismail *et al.*(2018)	BIM/IFC	Graph database	• Difference in output database type

1.2.3 Accessibility assessment

According to the characteristics of the target pedestrian, previous studies have extracted accessible route/area based on the presence of obstacles (Ge *et al.*, 2015; Karimi & Ghafourian, 2010; Kostic & Scheider, 2015; Yang *et al.*, 2020). Yang *et al.* (2020) focused on extracting accessible areas and routes, rather than the quantitative evaluation of accessibility. They proposed an accessible area generation approach with buffer zones. This approach considers distance constraints for indoor environments using a hybrid spatial data model. However, the accessible area has been determined based on distance, and obstacle information has not been considered in their approach. Karimi & Ghafourian (2010) proposed a new routing technique that satisfies users' special needs and preferences by considering accessibility. Factors for the building elements presented in the Americans with Disabilities Act (ADA) standard are classified into accessible routes and accessible Points of Interest (POI). The accessibility is determined as 'True' when each factor is satisfied associated criteria.

Other studies have derived factors affecting accessibility or have quantified accessibility (Bendel & Klüpfel, 2011; Church & Marston, 2003; Dao & Thill, 2018; Hashemi, 2018; Hashemi & Karimi, 2016; Vanclooster *et al.*, 2012). If accessibility is calculated quantitatively and used as cost, more sophisticated routing can be realized. Therefore, several studies have developed indices to present accessibility as quantitative values. Hashemi & Karimi (2016) and Hashemi (2018) developed an indoor accessibility index focusing on egressibility. This index quantitatively represents the effort required to arrive at a building's entrance. Moreover, they presented an evacuation routing algorithm based on calculated accessibility using the designed indices. As this study only focused on American regulations, however, more related regulations need to be reviewed to diversify the criteria. Furthermore, it is necessary to automate the creation of indoor graph models to apply accessibility indices. Meanwhile, Dao & Thill (2018) developed a tool for indoor layout design with high accessibility in terms of indoor travelers. The tool consists of an indoor transportation network modeler, accessibility evaluator, and audit component with visualizer and scenario builder. However, it did not include a detailed description of the network creation, and the structural information of each facility was not reflected when evaluating accessibility. In addition, other studies (Church & Marston, 2003; Kostic & Scheider, 2015) have limitations in that they excluded vertical transitions, reduced the details of complex spaces such as corridors, or focused on single floors only.

In conclusion, though accessibility information should be reflected in PWMD navigation by generating it using existing indices, existing indices tend to be mainly concentrated on a specific standard. Therefore, it is necessary to improve these indices by additionally investigating and reflecting various international regulations. Indices should reflect more relevant factors and have weights according to influences. Moreover, they should be improved in consideration of complex structured multi-floor indoor environments. Simultaneously, efficient generation of an adequate indoor database to be combined with accessibility information should be accompanied. [Table 1-3] presents a summary of major related studies of accessibility assessment.

[Table 1-3] Summary of major related studies of accessibility assessment

Previous studies	Target	Key issue of approach
Karimi & Ghafourian(2010)	Extraction of accessible route or area	• Use of criteria in ADA standard
Ge *et al.*(2015)		• Use of static and dynamic semantics
Kostic & Scheider(2015)		• Grid-based accessibility graph generation
Yang *et al.*(2020)		• Considering distance constraints without obstacle information
Hashemi & Karimi(2016), Hashemi(2018)	Deriviation of influential factors in accessibility assessment	• Using criteria in ADA standard • Proposal of accessibility index
Dao & Thill(2018)		• Proposal of accessibility evaluator for layout design

1.3 Research scope and flow

This study aims to develop a technique for generating a graph database that can be used directly for PWMD indoor navigation using scanned floor plans. The target database mainly involves geometric and topological space representations and stores accessibility information as semantic.

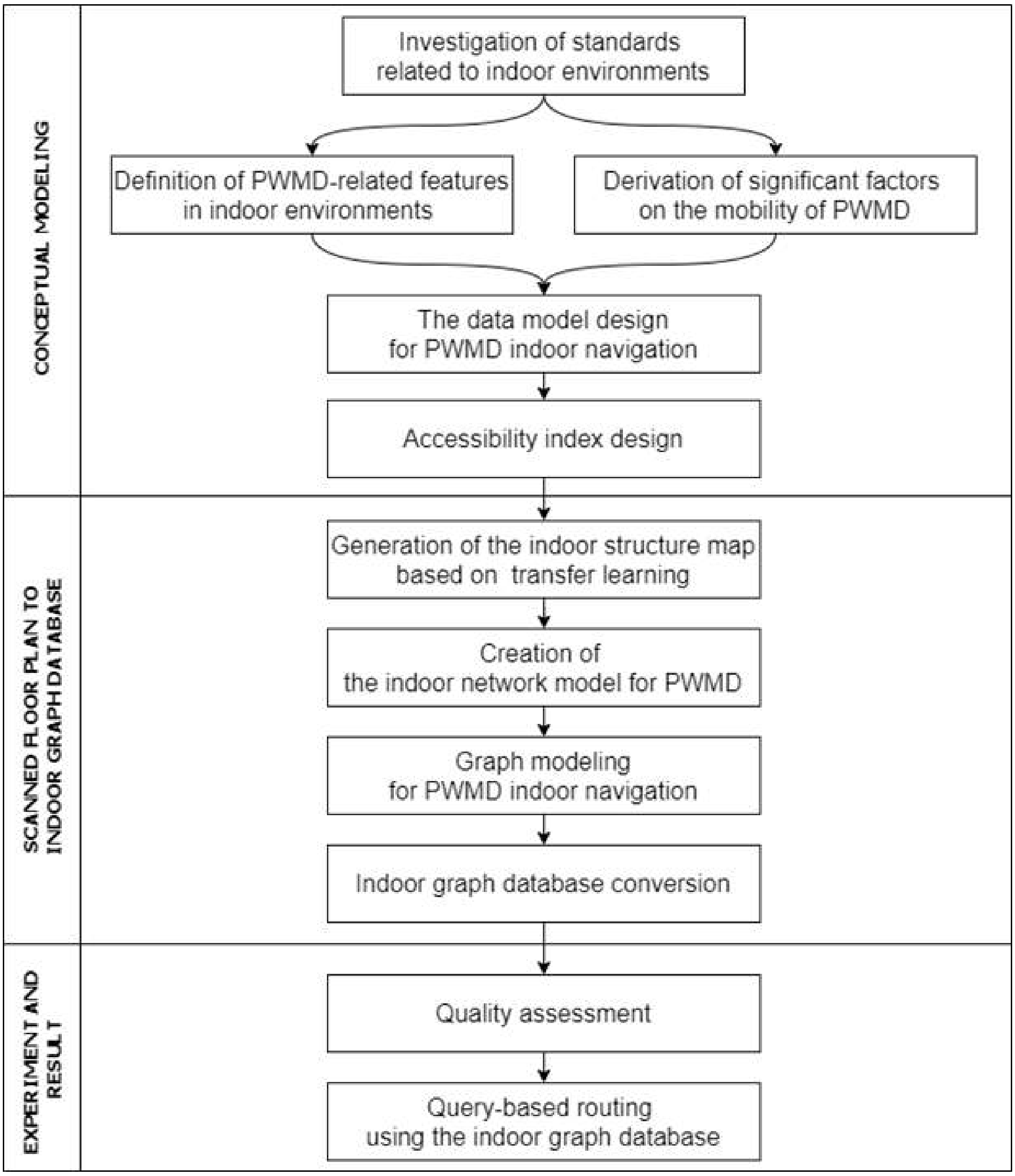
The term "PWMD" refers to individuals who cannot walk or have difficulty in walking, thus using mobility aids such as crutches, walkers, or wheelchairs (Department of Justice, 2010; Rimmer *et al.*, 2005). Also, "PWMD" includes people with physical impairments related to moving difficulties (UNHCR, 2019). In the relevant law of Korea, "mobility disadvantaged persons" are persons who feel inconvenient in mobility, such as persons with disabilities, the aged, pregnant women, persons accompanied by infants, and children (Korea Parliament, 2020). In consideration of these definitions, "PWMD" in this study is defined as limited to a mobility-impaired person who cannot travel by themselves, and a person who has cognitive and visual impairments is excluded. Nevertheless, the resultant database of this study can also be used for persons such as children, the elderly, and pregnant women who do not have any physical disabilities but require special needs for travel.

The information represented by the indoor spatial model is primarily classified as the geometric, topological, and semantic model (OGC IndoorGML, 2020). The indoor network model represents a path in a building, using a graph representation. It is a kind of topological model that includes connectivity information between indoor elements. Unlike a basic topological model, a network model simultaneously presents geometric information such as coordinates (Karas *et al.*, 2006).

However, in this study, the indoor network model is created as an intermediate output by prioritizing its efficient conversion to a graph database. Therefore, the model's scope is limited to partially omit geometric details, and more focus was devoted to retrieving topology.

This study comprises three parts. First, influential features and factors are derived in terms of PWMD. Based on corresponding significant features and their factors, a data model for PWMD indoor navigation is developed through the conceptual modeling of an indoor environment. Also, the accessibility index is designed based on the features and factors in the data model. Second, a process is developed for generating an indoor graph database using scanned floor plan images. Based on the designed data model, the target database involves major indoor features that can be extracted from floor plans. In the process, an indoor structure map can be extracted through a transfer learning-based approach, and an indoor network model can be configured with nodes and links through topology retrieval. Using the designed index, the accessibility of each feature is assessed to configure the feasible network model for PWMD. In this stage of accessibility assessment, detailed geometric characteristics of indoor spaces and installations should be investigated to calculate accessibility by applying the proposed index. The investigation requires a precise level of geometric information, thus assuming to be conducted manually, such as field survey. Moreover, a process is developed for converting the network model into a graph database that can be used for not only spatial routing but also other types of routing, such as semantic routing. Third, after creating actual data by applying the proposed process to test scanned floor plans, the appropriateness of the process is verified through a quality assessment of the experimental results. Furthermore, the utility of the proposed approach is examined through scenario-based routing using the generated indoor graph database. [Figure 1-2] shows the research flows.

The remainder of this paper is organized as follows. Section 2 presents the data model for PWMD indoor navigation, which was developed with the influence features. Also, the accessibility index designed based on the data model is introduced. Section 3 details the proposed process for generating an indoor graph database through structure and topology retrieval. The experimental results are analyzed in section 4, and section 5 describes the conclusions and limitations of this study.



[Figure 1-2] Research flow

2. Conceptual modeling

Indoor spaces can be expressed in various ways through the purposes of the spaces or the characteristics of the moving object (Li & Lee, 2013). In the case of PWMD, who require special needs when traveling, the accessibility of an indoor space is determined by the components of the building. For example, a wheelchair user cannot enter an inner space that has a doorway with a large sill, and a high cost is required for them to pass along a corridor with a steep slope. Wang & Niu (2018) insisted that data models should include relevant information to reflect this complex environmental context. Consequently, to assure the mobility of PWMD within an indoor environment, database for routing should be developed based on a data model that describes customized spaces for PWMD, not those for general pedestrians. As addressed in 1.2.1, although several previously developed data models describe the indoor environment, they do not include detailed feature modeling with mobile constraints considering PWMD applications. A model with an adequate representation of indoor environments for specific purposes may induce promoting relevant services. In this section, a data model for PWMD indoor navigation is designed by modifying the data model of Park *et al.* (2020).

Service developers do not need to define separate data models or build individual data in implementing various applications using shared data sets generated based on the standardized data model (Iida *et al.*, 2015). Thus, a standardized data model is helpful for various applications. Through definining the data model as an extended module of IndoorGML, the standardized application schema of indoor spatial information, the target database based on the corresponding data model has the benefit of sharing data easily for PWMD indoor navigation.

Therefore, significant spaces and facilities that affect the travel of PWMD were defined as indoor features within this proposed data model, based on elements of the IndoorGML core and navigation module. Detailed attributes of each feature in the designed data model are set for each feature by referring to relevant local and international indoor space regulations.

2.1 Relevant features and factors

The ADA standard (Department of Justice, 2010) defines requirements to increase the accessibility of spaces in building considering the special needs of the mobility and visually impaired (Karimi & Ghafourian, 2010). Using the ADA standard, previous works (Hashemi, 2018; Hashemi & Karimi, 2016; Karimi & Ghafourian, 2010) have derived factors affecting PWMD's mobility for indoor features; these studies used corresponding factors in evacuation routing. Similar to the ADA standard in the United States, other countries have regulated a building design to improve the PWMD's mobility in an indoor environment. Specifically, the Barrier-Free Certification System (BFCS) is implemented in Korea, and the criteria for scoring the accessibility of each space are presented in the system. BFCS aims to assure accessibility and convenience for PWMD by removing fundamental obstacles (Lee, 2012; Lee *et al.*, 2010).

As in previous research, relevant regulations can be used as basic information to evaluate the accessibility of indoor spaces. Spaces and facilities that can affect the mobility of PWMD are determined as indoor features by investigating ADA, BFCS, and related international regulations[3]. Also, characteristics that may act as constraints for each feature are derived as attributes (factors) of indoor features. [Table 2-1] shows the results of analyzing features and factors that influence the mobility of PWMD in indoor environments.

3) The criteria introduced in previous studies (Lee *et al.*, 2010; Lee, 2012) that compared each country's standards(welfare village in Japan, CSA-B651-04 in Canada, DIN 18024/18040 in Germany, SN 521 500 in Switzerland, Part M in the UK, and AS 4299 in Australia) were mainly used.

[Table 2-1] Influential factors of PWMD indoor mobility

Feature	Factors	ADA	BFCS	Others*
Corridor	width(hallway width, turning width)	●	●	●
	slope(running, cross)	●	●	●
	changes in level(vertical, beveled)	●	●	
	raised spot on the floor		●	●
Stair	width	●	●	●
	riser	●	●	●
	tread	●	●	●
	turning width	●		
	with ramp	●		
	raised spot on the floor		●	●
Door	width	●	●	●
	height	●		
	opening force	●		
	distance of parts(handles, closer, stop)	●		
	maneuvering clearance	●		
	manual doors(sliding, folding)	●		
	automatic doors	●		
	revolving doors/gates/turnstiles	●		
	door sill		●	●
	front distance	●	●	●
	parallel distance	●		●
Elevator	area (shorter and longer dimension)	●	●	●
	front space		●	●
	passing width	●	●	●
Room	shorter/longer dimension	●	●	●
	raised spot on the floor		●	●

* 'Others' column represents common criteria in related regulation of other countries

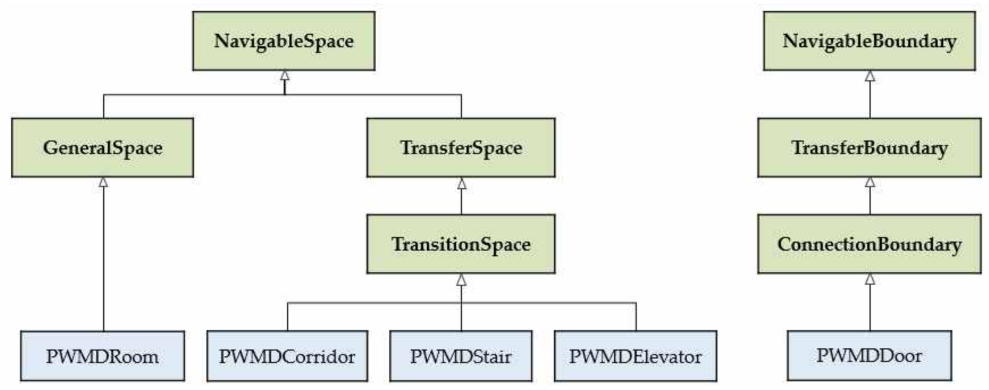
2.2 Proposed data model

To design a data model that reflects the PWMD-related information derived by examining various standards as shown in [Table 2-1], significant features and factors commonly included in those regulations are extracted and categorized ([Table 2-2]). When it is necessary to distinguish the type of feature, the 'type' attribute is added. In addition, for features that can guarantee accessibility using auxiliary facilities such as wheelchair lifts, the 'with facilities' item is appended as the attribute. The features are also mapped with associated elements in the IndoorGML navigation module. Transitions between floors are generally accomplished through stairs, elevators, and escalators. PWMD has many vertical transition restrictions, including inter-floor travel, so the proposed data model should include such spaces and facilities related to vertical movement as essential features. Unlike stairs and elevators, escalators are contained as installation features in the proposed data model. The door is also a key feature for mobility, as are stairs and elevators for vertical movement, corridors for horizontal movement, and rooms for activities. Access to a space through a door is determined by corresponding door's characteristics, such as its width and the presence of a door sill (Fleiner *et al.*, 2017; Hashemi, 2018; Hashemi and Karimi, 2016; Karimi & Ghafourian, 2010; Kostic & Scheider, 2015). Therefore, factors of the door such as 'type', 'width', 'front and parallel distance', and 'door sill' are defined as attributes of the *PWMDDoor* feature.

[Table 2-2] Significant features and their attributes

Feature	Related feature of IndoorGML	Attributes
Corridor	*TransitionSpace*	width(hallway width, turning width), slope(running, cross), changes in level(vertical, beveled), raised spot on the floor
Stair	*TransitionSpace*	with wheelchair lift, with ramp
Door	*ConnectionBoundary*	width, front/parallel distance, door type, automatic doors, door sill
Elevator	*TransitionSpace*	shorter dimension of car, passing width, shorter dimension of front space
Room	*GeneralSpace*	type, shorter dimension, raised spot on the floor

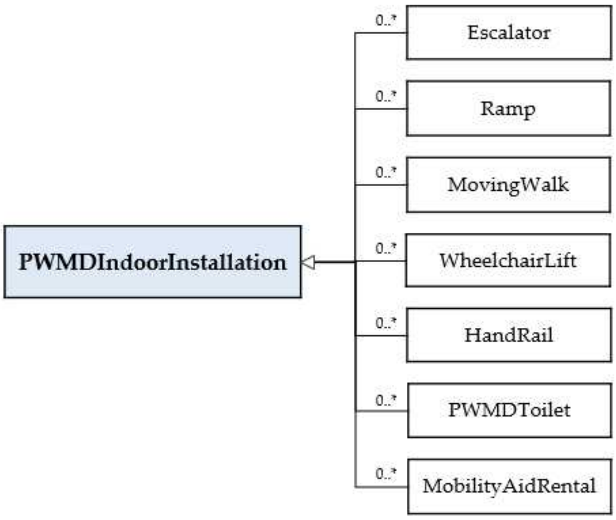
The IndoorGML core module expresses indoor spaces by focusing on geometric and topological characteristics. Furthermore, its navigation module describes the indoor space focusing on routing (Kang & Li, 2017). In both modules, the indoor environment is expressed using two major features: space and boundary. Spaces that people can access are defined as *NavigableSpace* in the IndoorGML navigation module. *NavigableSpace* features include a *GeneralSpace* for specific activities and a *TransferSpace* for transition between spaces (Li & Lee, 2013). In this study, the subclasses of *GeneralSpace* and *TransferSpace* are redefined in terms of PWMD mobility ([Figure 2-1]). *PWMDRoom* is added as a sub-feature of the *GeneralSpace* of the navigation module. The three other space features presented in [Table 2-2] are connected with *TransitionSpace* as sub-features, and the corresponding *TransitionSpace* is the child feature of *TransferSpace.* Specifically, in the proposed model, *TransitionSpace* is inherited by *PWMDCorridor* for horizontal movement and by *PWMDElevator* and *PWMDStair* for vertical. The attributes of each feature are represented in [Table 2-2].



[Figure 2-1] Subclasses of navigable spaces and boundary for PWMD

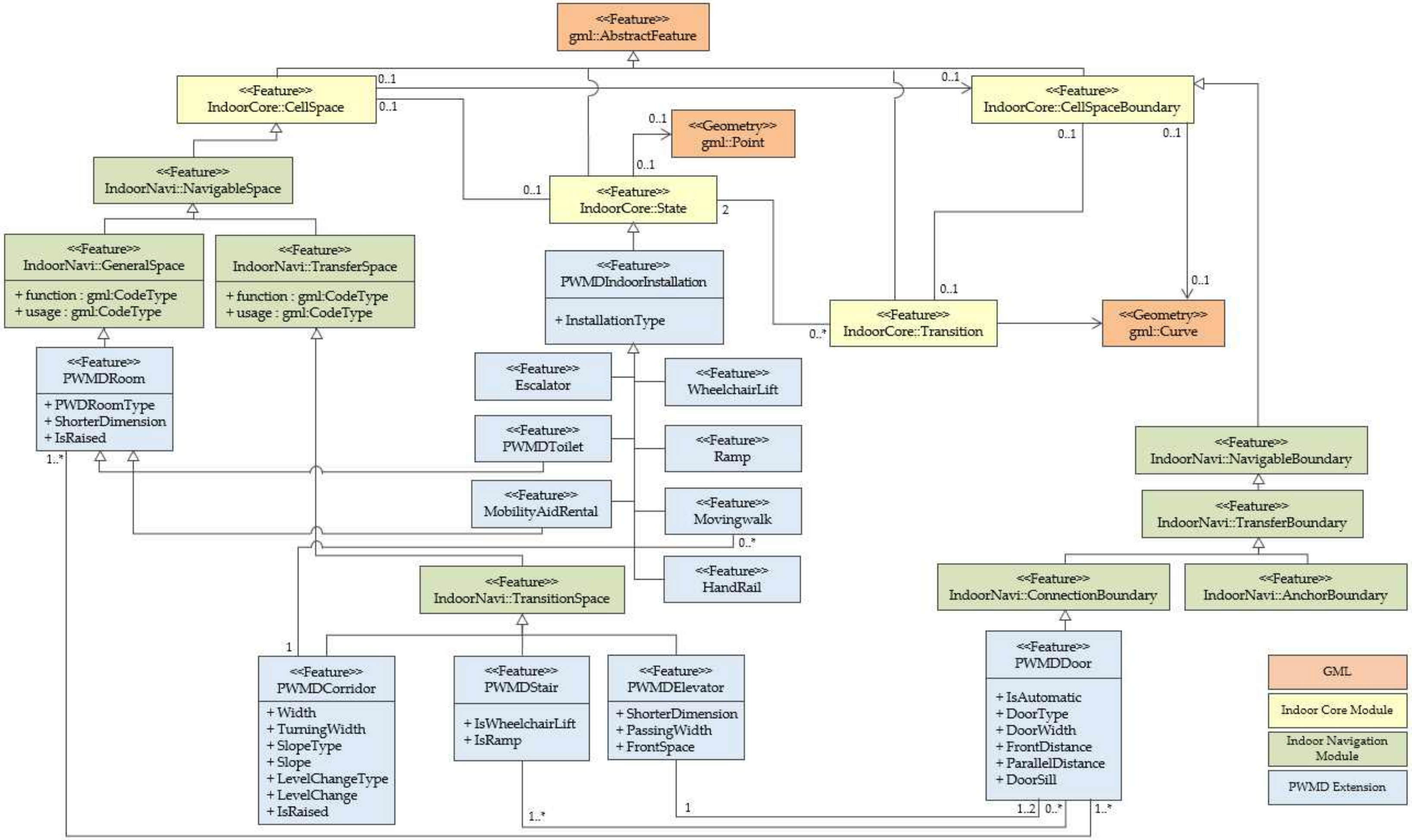
The inner entrance of each space is defined as a *ConnectionBoundary* feature in the IndoorGML navigation module. The accessibility of PWMD to each room is affected not only by the space characteristics, but also by the properties of the associated doorway. Therefore the *PWMDDoor* can be added as a subclass of the *ConnectionBoundary* feature.

There are specific facilities and areas for supporting PWMD in buildings. A wheelchair rental shop or a wheelchair lift installed on a flight of stairs are corresponding examples. Facilities or areas for such a particular purpose need to be involved as PWMD-related installations in indoor routing services. In this context, seven features: *WheelchairLift, HandRail, MovingWalk, Ramp, Escalator, PWMDToilet,* and *MobilityAidRental* are defined as the *PWMDIndoorInstallation* feature ([Figure 2-2]). In particular, *Ramp* and *Escalator* are added as installation features, since they are spaces for vertical transition and concurrently serve as facilities for supporting travel. When constructing an indoor model, *PWMDToilet* is not only classified as *PWMDRoom* (as a general space), but also needs to be emphasized as a space with a special purpose.



[Figure 2-2] Indoor installation features of PWMD

Unified Modeling Language (UML) class diagrams of the model designed by integrating features of indoor spaces, boundaries, and installations for PWMD as defined above are presented in [Figure 2-3]. The newly defined sub-features of *NavigableSpace* are connected with the *CellSpace* feature of the core module. Similarly, the *CellSpaceBoundary* of the core module is inherited by *NavigableBoundary,* which includes the added *PWMDDoor* feature. Also, *PWMDInstallation* is defined as a subclass of *State* in the core module.



[Figure 2-3] UML diagram of PWMD model

Similar to the ‘function’ and ‘usage’ attributes of the *CellSpace* feature in the IndoorGML navigation module, the ‘type’ attributes of *PWMDRoom, PWMDCorridor,* and *PWMDDoor* can use predefined values. The classification for each type is shown in [Table 2-3]. ‘PWMDRoomType’ is the classification of a general space in consideration of the available navigation support for PWMD. The ‘Activity room’ class includes all areas for normal activities.

[Table 2-3] Type attributes of features

Feature	Attribute	Enumeration
PWMDRoom	PWMDRoomType	Activity room
		Toilet
		Mobility aid rental
PWMDCorridor	LevelChangeType	Vertical
		Beveled
		None
	SlopeType	Running
		Cross
		None
PWMDDoor	DoorType	Sliding
		Hinged
		Folding
		Revolving

2.3 Space accessibility for PWMD

In this study, an accessibility index is designed to generate accessibility information for the target indoor features in the network model in terms of PWMD. In IndoorGML (OGC IndoorGML, 2020), 'accessibility', defined as indoor semantics, represents the feasibility of entering or passing each space under constraints. According to the example presented in the IndoorGML document, if the moving object has a constraint of 1.2m width, a room having a door width smaller than 1.2m is not accessible. Providing accessibility information for each indoor space or POI can greatly assist PWMD when planning activities in a given building (Karimi & Ghafourian, 2010). Most previous studies into determining routing for PWMD aimed to extract accessible routes by examining whether they are accessible based on obstacles. However, for a detailed routing, it is necessary to quantitatively assess accessibility and reflect it as cost in the path computation with network models.

In general, either relevant documents (Hashemi, 2018; Hashemi & Karimi, 2016; Karimi *et al.*, 2014; Karimi & Ghafourian, 2010; Kostic & Scheider, 2015) or crowdsourced knowledge (Qin *et al.*, 2016; Onorati *et al.*, 2014) can be used to generate accessibility information. When assessing accessibility, it is necessary to document a walking path's characteristics, including obstacle information (Qin *et al.*, 2016). Hashemi (2018) reviewed the ADA standards and derived accessibility requirements for PWMD for corridors, stairways, doorways, ramps, and elevators. An equation has also been designed to calculate accessibility quantitatively. By referring to the accessibility index defined in previous studies, related features and factors are adjusted based on the proposed data model in this study. Also, indices are improved by reflecting additional criteria from BFCS. A survey had been conducted on the PWMD before establishing accessibility criteria in BFCS. Therefore, user experience (PWMD's experience) can be reflected in the accessibility index by referring to criteria presented in BFCS. Specifically, scores are assigned by factors in the BFCS, and the scoring scale is determined according to the influence of each factor. That is, a factor with a higher score means a more critical factor. Therefore, the accessibility index can be improved by assigning the weights for each factor based on the BFCS scoring system.

2.3.1 Influential factors within indoor environments

The features and factors (influential attributes) constituting the indoor model for PWMD navigation are derived by examining relevant standards and regulations ([Table 2-2]). Additionally, a particular feature (toilets for PWMD) and its attribute are added with distinct criteria.

- Corridors: width (hallway, turning), slope (running, cross), changes in level (vertical, beveled), raised spot on the floor
- Stairways: with wheelchair lift, with ramp
- Doorways: width, front distance, parallel distance, revolving door, automatic door, door sill
- Elevators: shorter dimension of car, passing width, shorter dimension of front space
- Room: shorter dimension, raised spot on the floor
- Toilet: passing width, slope, connected corridor width, raised spot on the floor

[Table 2-4] illustrates the results of investigating the criteria, indicating the level of difficulty in access for the above features and factors. However, criteria of accessibility for each factor may vary depending on the detailed type and degree of disabilities. Therefore, the criteria for people with wheelchairs are prioritized to design a rigorous index as representative in this study.

The second to fourth columns in the [Table 2-4] represent the quantitative thresholds in ADA and BFCS for each indoor feature (space) to be considered accessible. For example, the width of a door is essential to determine the accessibility of entering the area it serves. According to the third and fourth columns, while a door with a width of >815 mm is considered accessible by ADA, BFCS uses a width threshold of >900 mm. As mentioned earlier, BFCS includes a scoring system that reflects weights for each criterion through investigating user experiences, unlike ADA. The proposed index in the following section considers the scores assigned for each factor of BFCS as a weight (the fifth column in [Table 2-4]).

[Table 2-4] The criteria indicating level of difficulty for access

Feature	Factor	ADA(mm)	BFCS(mm)	Weight	Variable
Corridor	width	>915	>=1,200	3	C_1
	turning width	>1,220		1	C_2
	running slope	<0.05	<=0.0833	3	C_3
	cross slope	<0.0208		1	C_4
	vertical level change	<6.4		1	C_5
	beveled level change	<13		1	C_6
	with raised spot			2	C_7
Stair	with wheelchair lift			1	S_1
	with ramp			1	S_2
Door	width	>815	>900	3	D_1
	front distance	>1,525	>= 2,100	3	D_2
	parallel distance	>455	>=600	1	D_3
	door sill	<13	<20	3	D_4
	is revolving door	not accessible		1	D_5
	is automatic door			1	D_6
Elevator	shorter dimension of car	>1,354		1	E_1
	shorter dimension of front space		>=1,400	2	E_2
	passing width		>=800	2	E_3

Feature	Factor	ADA(mm)	BFCS(mm)	Weight	Variable
Room	shorter dimension	>=915	>=1,200	3	R_1
	with raised spot			2	R_2
Toilet	passing width	>815	>=900	3	T_1
	beveled level change - slope		<=0.0833	3	T_2
	connected corridor width		>=900	3	T_3
	with raised spot			4	T_4

* Gray cells represent each criterion used in index design

2.3.2 Accessibility index

As presented in [Table 2-5], the accessibility index is designed based on the criteria shown in [Table 2-4] to indicate the difficulty of accessing spaces. The variables in the second column in [Table 2-5] represent each significant factor. The designed index in the third column is based on the quantitative criteria specified by ADA and BFCS. In addition, the weight in the fifth column of [Table 2-4] is reflected in the proposed index. ADA and BFCS have different standards, as shown in [Table 2-4]. In this study, the stricter of these two criteria are selected to be used in the equation. For example, C_1 represents the width of the corridor. A width of >915 mm is acceptable to access for PWMD by ADA, whereas BFCS defines corridors that are >1,200 mm in width as accessible. Thus, BFCS has the more rigorous criterion for corridor width. Therefore, the index for C_1 is proposed using BFCS criteria, and the threshold is calculated with ADA. An example of calculating the threshold for corridor width is as follows:

$$(915/1200) \times 3 \fallingdotseq 2.29 \tag{2-1}$$

Features with larger index values have higher accessibility. Also, a feature with a calculated value under the threshold may be determined as inaccessible. The indices and thresholds are proposed primarily considering wheelchair users and can be subdivided according to the type and degree of disabilities.

The value calculated using the accessibility index can be regarded as a cost during path planning. However, accessibility is an attribute that other factors cannot supplement. For example, if a space has a very narrow width, it should be determined as an inaccessible space, even if other conditions such as slope and level change are acceptable.

Therefore, the thresholds for determining inaccessibility are presented by each factor.

[Table 2-5] The proposed accessibility index for significant features

Feature	Variable	Index	Threshold	Total index
Corridor	C_1	$(C_1/1200)\times 3$	2.29	$AC_{corr}=\sum_{i=1}^{7} C_i$
	C_2	$(C_2/1220)$	1	
	C_3	$(0.05/C_3)\times 3$	1.80	
	C_4	$(0.0208/C_4)$	1	
	C_5	$(6.4/C_5)$	1	
	C_6	$(13/C_6)$	1	
	C_7	$(1-C_7)\times 2$	2	
Stair	S_1	S_1	1	$AC_{stair}=\sum_{i=1}^{2} S_i$
	S_2	S_2	1	
Door	D_1	$(D_1/900)\times 3$	2.72	$AC_{door}=\sum_{i=1}^{6} D_i$
	D_2	$(D_2/2100)\times 3$	2.18	
	D_3	$(D_3/600)$	0.76	
	D_4	$(13/D_4)\times 3$	1.95	
	D_5	$(1-D_5)$	1	
	D_6	D_6	-*	
Elevator	E_1	$(E_1/1354)\times 2$	2	$AC_{elev}=\sum_{i=1}^{3} E_i$
	E_2	$(E_2/1400)\times 2$	2	
	E_3	$(E_3/800)\times 2$	2	
Room	R_1	$(R_1/1200)\times 3$	2.29	$AC_{room}=\sum_{i=1}^{2} R_i$
	R_2	$(1-R_2)\times 2$	2	
Toilet	T_1	$(T_1/900)\times 3$	2.72	$AC_{toilet}=\sum_{i=1}^{4} T_i$
	T_2	$(0.0833/T_2)\times 3$	3	
	T_3	$(T_3/900)\times 3$	3	
	T_4	$(1-T_4)\times 4$	4	

* 'with automatic door' is not an essential factor to determine accessibility (without threshold)

3. Indoor graph database for PWMD from scanned floor plans

3.1 Retrieving structure of indoor spaces

To generate a target database for PWMD navigation using the scanned floor plan as source data, retrieval of lost information is essential. Although pixels constituting indoor features can be distinguished in the scanned floor plan, the feature class is not involved in those pixels. Also, they do not include any information on adjacency and connectivity among features. Therefore, the previous methods of generating navigation databases from semantically rich source data such as BIM/IFC cannot be applied to the scanned floor plans. In this regard, the generation technique of graph database for PWMD, including the information retrieval, is proposed in this study. The first step in generating a target graph database from scanned floor plans involves structure retrieval.

In recent years, deep learning approaches have already achieved much success regarding floor plan analysis. In particular, various learning-based techniques have been used to extract geometric patterns from floor plan images. As indoor features such as walls and openings can be defined as targets for reconstructing the structures of rooms, many existing studies have proposed deep learning frameworks for extracting indoor features (Dodge *et al.*, 2017; Jang *et al.*, 2020; Kim *et al.*, 2021; Wu *et al.*, 2020; Zeng *et al.*, 2019). In other words, various deep networks have been used to extract features including walls and openings from scanned floor plans. The pre-trained deep networks for indoor geometric model creation can be reused to generate a feature map suitable for topology retrieval after retraining. However, the pre-trained networks have tended to focus on creating geometric models. The purpose of previous deep networks is to precisely reconstruct the geometry of indoor elements, thereby changing annotation is needed for retraining in consideration of topology retrieval.

In this study, a transfer learning-based approach is proposed for retraining a pre-trained model with changed annotations to retrieve the structure of space considering reconstruction of adjacency and connectivity between spaces. Specifically, an indoor structure map for topology retrieval is generated using an optimized model by applying transfer learning to a pre-trained model for geometry extraction. Transfer learning aims to share the trained knowledge of the entire network. This allows for better performance to be achieved using less training data and time. Transfer learning is to borrow meaningful information from related tasks for a new task, with an insufficient training set (Tan *et al.*, 2018).

3.1.1 Pre-trained model for detecting indoor geometry

In this study, a modified ResNet-based model (Ministry of Land, Infrastructure, and Transport, 2020) is used as a pre-trained model. The model was trained with the Seoul National University (SNU) dataset to extract the geometric information of indoor spaces from scanned floor plans. Since the newly defined task in this study is to extract indoor structural information for network model creation, the modified ResNet-based model trained with the SNU dataset, composed of large and complex buildings, is adopted as the pre-trained model. The target objects of the pre-trained model were defined with several conditions: 1) they are represented by a consistent symbol, 2) the number of objects illustrated in the training floor plans is sufficient, and 3) they are essential to reconstructing the indoor structure. The target objects determined under these conditions were walls, doors, windows, stairs, and elevators.

The pre-trained model adopted the approach of Generative Adversarial Network (GAN). The GAN consists of a generator and a discriminator. The generator creates an image close to the real, and the discriminator determines whether the generated image is fake. Therefore, the network is competitively trained to minimize the generator loss and maximize the discriminator loss in the approach of GAN. The generator was based on ResNet50 ([Table 3-1]), but was adjusted to extract the feature map in raster format using the deconvolution layer, instead of the last fully connected layer. As shown in [Table 3-1], the generator has been adjusted not to decrease in output size when passing through each layer to restore the final feature map to the same size as the input.

The SNU dataset consists of various sizes, ranging from a minimum of 1,271*1,271 to a maximum of 6,034*6,034. A fatal loss of geometric information occurs if high-resolution images are excessively reduced in size when being inputted into the network (Suh *et al.*, 2020). Furthermore, it is difficult to maintain the original shape of the object. Therefore, instead of reducing input floor plan images, the network was trained by dividing each image into patch of 512*512, with 30% overlap. Performance can be improved by combining the L1 distance to the generator loss, as the task of generator is not only fool the discriminator but also generate output similar to ground truth (Isola *et al.*, 2017).

The generator loss ($\mathcal{L}_g$) of pre-trained model is represented as :

$$\mathcal{L}_g = \lambda \cdot \mathcal{L}_{style} + \mathcal{L}_{L_1} \tag{3-1}$$

$$\mathcal{L}_{style}(G,D) = E_{x,y}[\log D(x,y)] + E_{x,z}[\log(1-D(x,G(x,z)))] \tag{3-2}$$

$$\mathcal{L}_{L_1}(G) = E_{x,y,z}[\, \| y - G(x,z) \|_1] \tag{3-3}$$

Where λ is style ratio, x is observed image, z is random noise vector, and y is mapped as $G:\{x,z\} \rightarrow y$. D is discriminator to determine whether the generated image is fake, and G is generator to generate an style transferred images that cannot be distinguished from real.

[Table 3-1] The architecture of the generator

Block name	Output size	50-layer
Block 1	256×256	$7\times7, 64$, stride 2 3×3 max pool, stride 2
Block 2	128×128	$\begin{bmatrix} 1\times1, 64 \\ 3\times3, 64 \\ 1\times1, 256 \end{bmatrix} \times 3$
Block 3	128×128	$\begin{bmatrix} 1\times1, 128 \\ 3\times3, 128 \\ 1\times1, 512 \end{bmatrix} \times 4$
Block 4	128×128	$\begin{bmatrix} 1\times1, 256 \\ 3\times3, 256 \\ 1\times1, 1024 \end{bmatrix} \times 6$
Block 5	128×128	$\begin{bmatrix} 1\times1, 512 \\ 3\times3, 512 \\ 1\times1, 2048 \end{bmatrix} \times 3$

3.1.2 Dataset with new annotation

As it is difficult to consider a particular situation such as evacuation through a window for PWMD, windows were not considered openings in the data model proposed in section 2. Therefore, windows are excluded among target objects; the remaining objects are identical to the target features of this study: wall (room), door, elevator, and stair.

In the pre-trained model, the SNU dataset was annotated to improve the accuracy of pixel-wise segmentation by maintaining the geometric shapes of walls. In contrast, the goal of the learning phase in this study is to extract indoor structural information for topology retrieval. Accordingly, new annotations were created focusing on the generation of indoor structure maps.

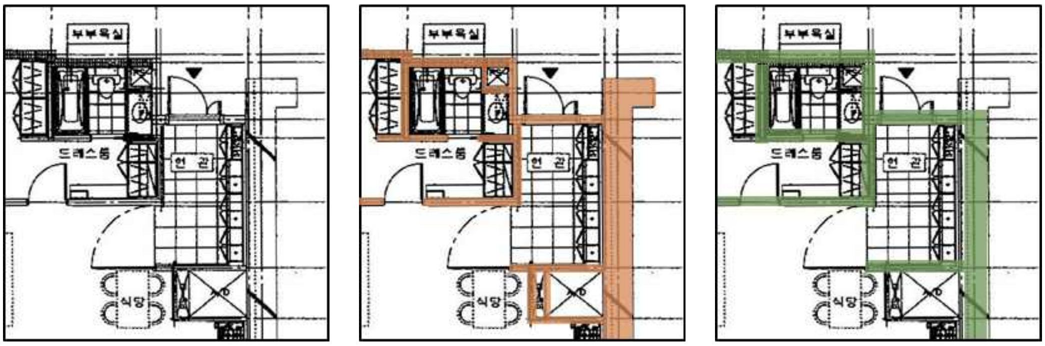
In this study, new datasets were created using part of the SNU dataset (Kim, 2020; Ministry of Land, Infrastructure, and Transport, 2019). The SNU dataset consists of scanned floor plans provided by the Facility Support Department; it includes 230 floor plan images. According to Kim (2020), the floor plans in the SNU dataset are characterized as follows: 1) the object building sizes are relatively large, and the structure of each building is complex compared to other datasets such as the CVC-FP[4)] and EAIS[5)], 2) floor plans are drawn by floor, 3) connections between floors are expressed through multiple elevators and stairs, and 4) the proportions of the target object areas, relative to the total area of each floor, are much smaller than those of other datasets, as the images have high resolutions (3,000 pixels or more). In summary, the SNU dataset is appropriate as a test set for

4) A dataset with four types of real floor plan documents with annotation; CVC-FP is acronym for Computer Vision Center-Floor Plan

5) A dataset composed of diverse architectural drawings which are downloaded from Electronic Architectural Information System with annotation

generating a database for navigation due to the characteristic that it is composed of floor plans for large and complex buildings drawn by every single floor.

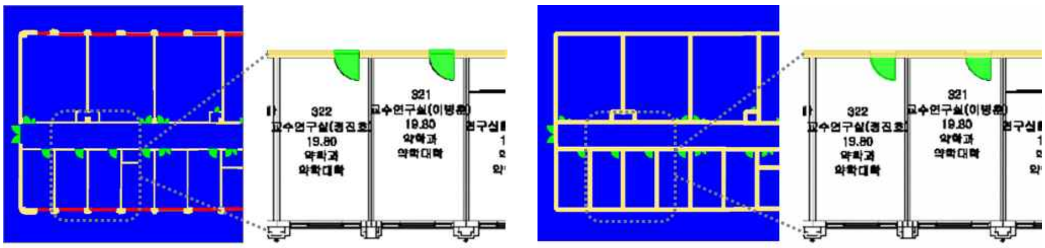
New annotations for retrieving the structures of indoor spaces were generated for several floor plans in the SNU dataset. Kim *et al.* (2021) proposed a unified style of annotation for the EAIS dataset to express the structures of indoor spaces. As shown in [Figure 3-1], wall annotations tend to focus on the division of rooms, not the geometric shapes of walls, following the unified style of Kim *et al.* (2021). Therefore, walls are recognized as objects for forming rooms in the training stage by representing them as expanded linear features.



(a) Architectural Floor plan | (b) Annotation with actual walls | (c) Annotation with simplified walls

[Figure 3-1] Differences in wall annotation described in Kim *et al.* (2021)

[Figure 3-1(b)] shows a detailed representation of walls, designed to preserve the actual shape, and [Figure 3-1(c)] describes a simplified representation of the walls regarding how they enclose each space. In the indoor network model, spaces are abstracted to nodes and links, so it is more efficient to express walls as shown in [Figure 3-1(c)]. Therefore, new annotations were generated by referring to those of Kim *et al.* (2021). Additionally, to extract the structures of spaces, the annotations of doors and walls were modified so that rooms could be enclosed only with walls ([Figure 3-2]). [Table 3-2] shows a part of the newly annotated dataset in this study, compared to that of the existing SNU dataset.

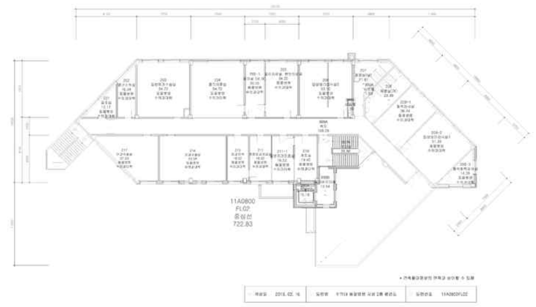
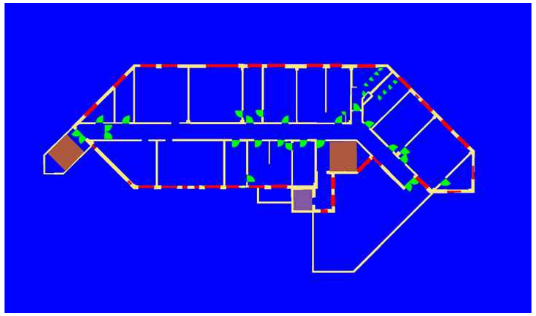
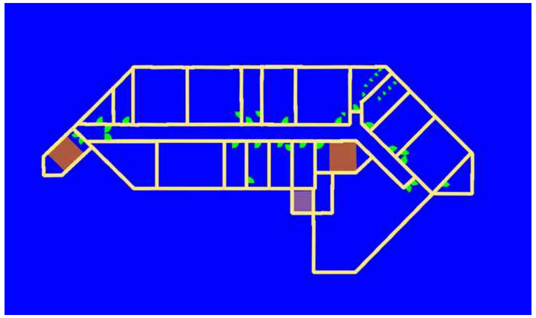
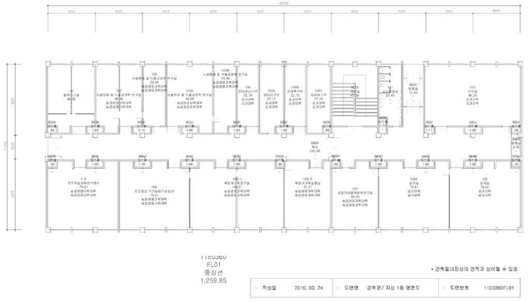
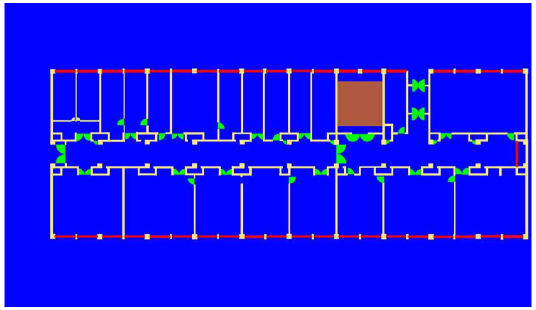
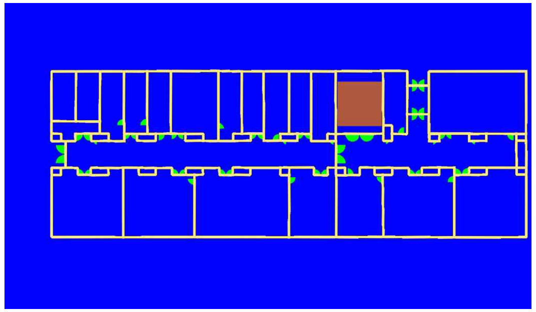


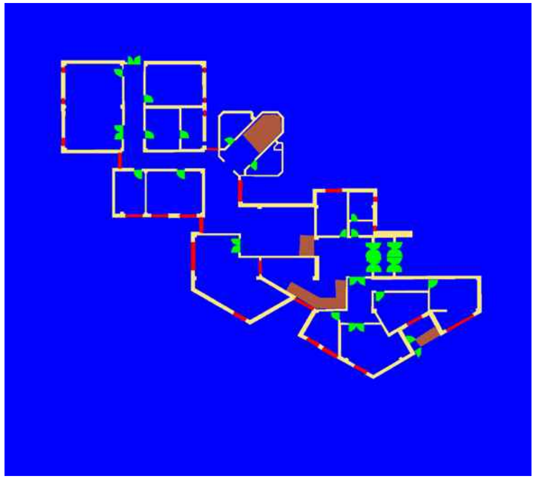
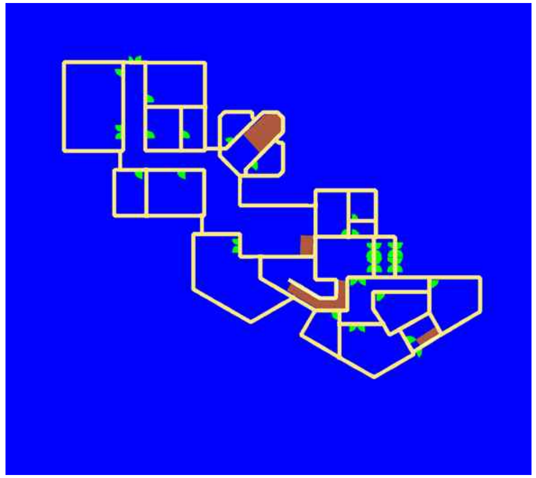
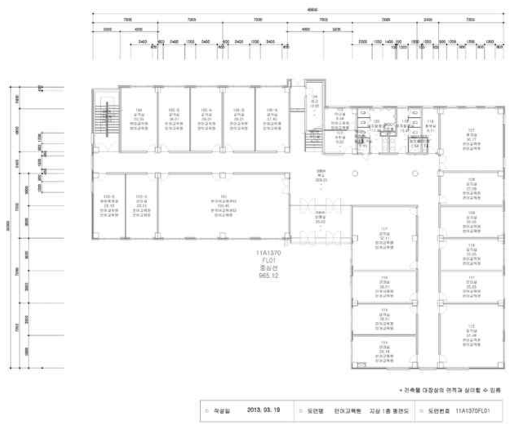
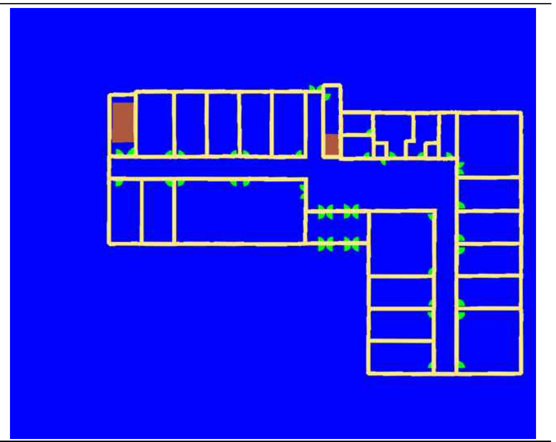
(a) Annotation for the pre-trained model

(b) Modified annotation for transfer learning

[Figure 3-2] Modification of door and wall annotation

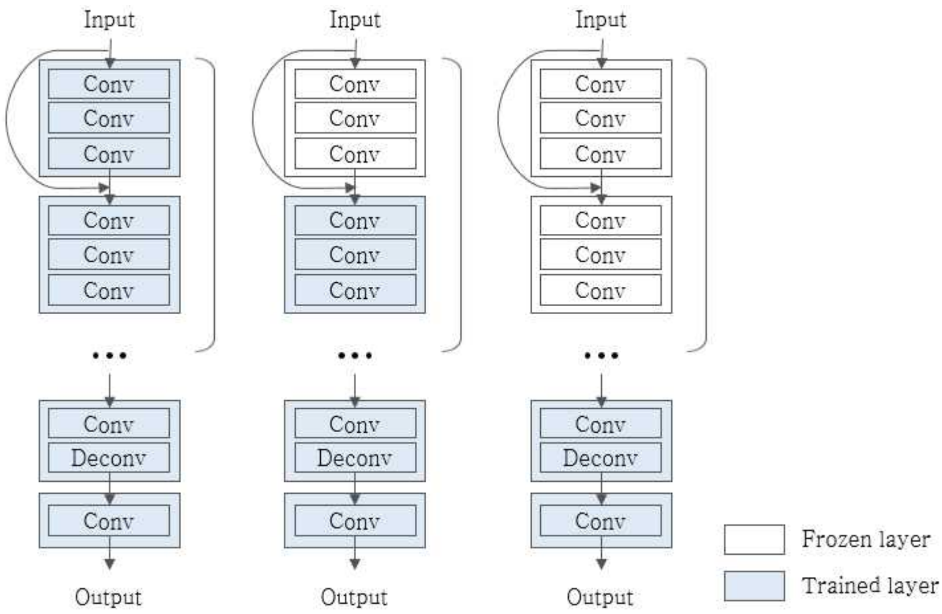
[Table 3-2] Newly annotated dataset in this study

Input floor plan	Original annotation for pre-trained model	New annotation in this study
		
		

Input floor plan	Original annotation for pre-trained model	New annotation in this study
		
		

3.1.3 Transfer learning-based approach

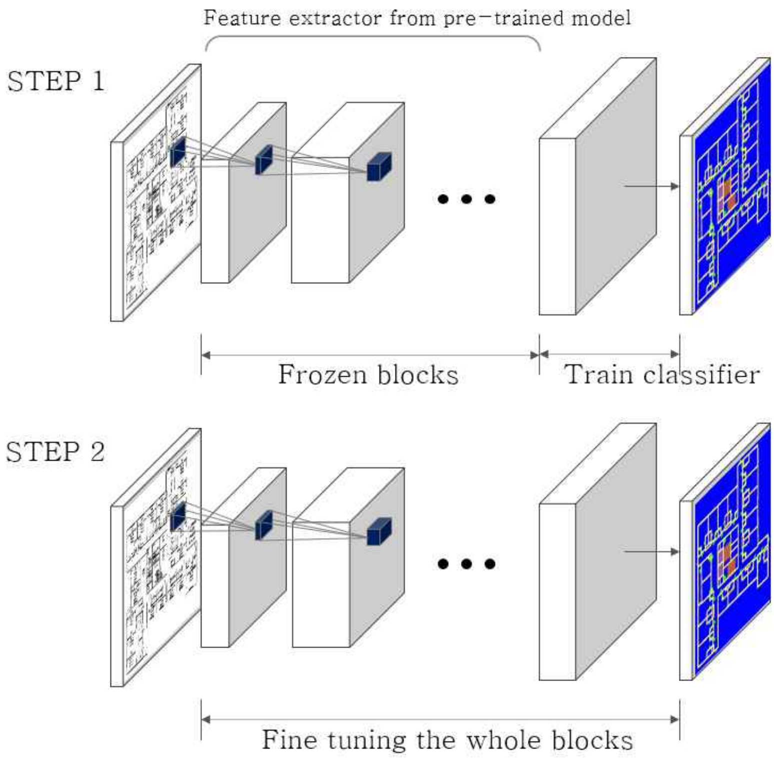
Several things must be decided to perform transfer learning effectively: the knowledge to be transferred, the learning algorithm used to transfer knowledge, and the proper cases used to apply transfer learning (Pan & Yang, 2009). Fine-tuning is required to adjust the pre-trained model to fit the new task. The dataset of the pre-trained model and the new task can be regarded as the source and the target, respectively. As shown in [Figure 3-3], the optimal fine-tuning approach depends on the decision of the layer to start training in the pre-trained network, according to the amount and similarity of the target datasets. If there are enough target datasets, it is helpful to train the entire network by fetching only the structure of the pre-trained model. On the other hand, if the target dataset is extremely small and its similarity to the source dataset is high, it is appropriate to retrain only the classifier, while using the feature extractor of the pre-trained model. In other words, the layers that need to be retrained can be determined according to the target dataset volume and its similarity with the source dataset.



[Figure 3-3] Various fine-tuning approaches

The pre-trained model was trained using 198 floor plans in the SNU dataset. The new annotated dataset comprises 32 floor plans which were not used in training phase of the pre-trained model. As explained earlier, if the amount of new data is small, back propagation for the entire model should be avoided, as this would cause over-fitting. However, if the characteristics of target data are similar to the source data, the feature extractor can be transferred from the pre-trained model. In this case, the layers of the feature extractor are fixed during training, and only layers of the classifier are retrained. Although fast learning is possible through only training the classifier, the feature extractor and the classifier are independently trained in this approach. This independent training reduces the model's discriminative ability regarding new tasks (Zhao *et al.*, 2017). In order to solve this problem, an approach has been developed that includes the additional fine-tuning of all or part of the layers after training the classifier (Castelluccio *et al.*, 2015; Zhao *et al.*, 2017).

Indoor structure maps are created through two stages of transfer learning in this study ([Figure 3-4]). In the first stage, layers from blocks two to five, which are used as a feature extractor for indoor objects, in the pre-trained model are frozen, and these features are used to feed the latter layer block of the classifier. Then, the remaining blocks as a classifier are retrained with the new dataset. For the second stage, the feature extractor and the retrained classifier, which are trained independently, are integrated by fine-tuning.



[Figure 3-4] The steps of transfer learning

As mentioned above, the new annotations were made considering walls as objects for space division. Therefore, it is more important to extract walls with a unified style than to obtain exact walls with pixel-wise accuracy. Fine-tuning is performed by assigning weights to the style loss so that walls can be recognized as standards for enclosing rooms in the training stage. The extracted feature map is separated into four layers for each class (wall, door, stair, and elevator). This map is used to create a network model, as described in the following section.

3.2 Generating the indoor network model for PWMD

In this study, a technique for creating an indoor network model is proposed using the indoor structure map generated in section 3.1. The indoor network contains two primary components: nodes and links (Teo & Cho, 2016). A model with nodes and links is a general way to represent relationships between objects. As discussed in section 1.2, many previous studies have dealt with generation techniques for indoor network models. However, the target input of most previous techniques (Khalili & Chua, 2015; Mortari *et al.*, 2019; Taneja *et al.*, 2011; Taneja *et al.*, 2016; Teo & Cho, 2016; Yang & Worboys, 2015) is the file in a vector format, such as BIM/CAD, IFC, and OpenStreetMap (OSM). These files are preferable to use as source data for topology computation. Also, those sources include various semantic information for classifying types of indoor features and searching adjacent features. On the other hand, the target input of the proposed technique is an indoor structure map that only stores feature class values in pixels. Therefore, previous methods cannot be directly applied to an indoor structure map.

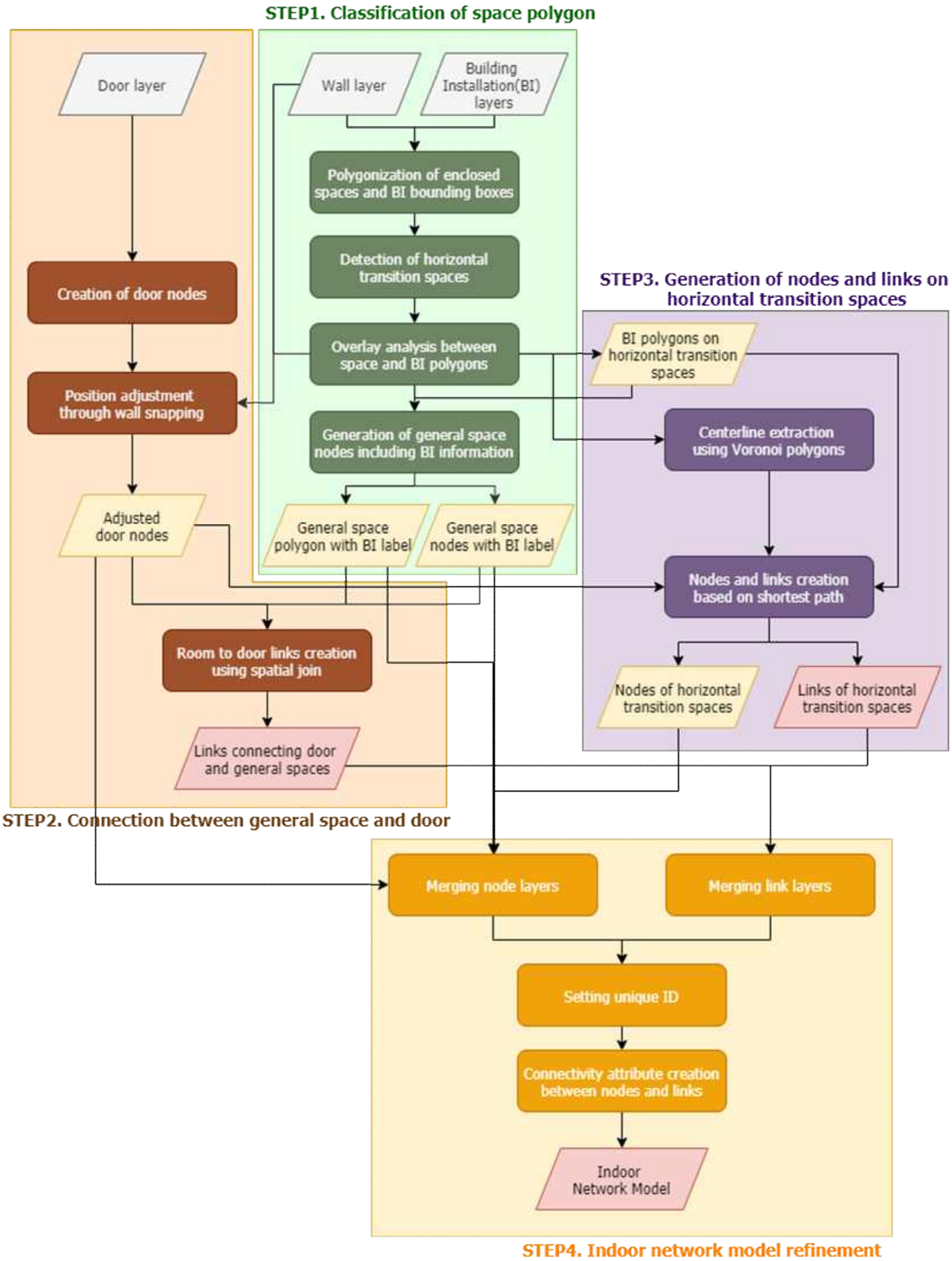
The methodologies of several previous studies using raster data also have limitations. Karas *et al.* (2006) automatically extracted indoor networks from architectural plans as blueprints by applying an image processing method. However, this method is limited in that input plans must be drawn in a restricted format. Moreover, the method proposed by Niu & Song (2019), which uses raster maps extracted through deep learning approach as source data, is inappropriate for creating a navigable network as it expresses the adjacency of the space without considering connectivity or accessibility.

In this context, an automatic process applicable to indoor structure maps is developed to create a network model referring to ideas introduced in previous studies. The proposed technique is differentiated in that it creates a network model suitable for navigation applications after restoring association between indoor features for topology retrieval. Also, the sub-procedures are developed by subdividing types of relation between features. For the creation of the indoor horizontal network model, the proposed process consists of four steps, as shown in [Figure 3-5].

- Step 1: As the first step in creating a network model, areas enclosed by walls are classified as horizontal transition spaces (so-called corridors) and general spaces. The inner centroids of general spaces are defined as nodes and stored with labels. Separate labels are given to general spaces on which building installations (elevators and stairs) are located through overlay analysis.
- Step 2: Door nodes are created using the extracted door pixels, and the door positions are adjusted by snapping them to the wall. Afterwards, door nodes are joined spatially with the space polygons to search associated general spaces. Then, links connecting the doors and associated general spaces are created.
- Step 3: Nodes and links for horizontal transition spaces are created in this step. The horizontal transition spaces detected in step 1 consist of relatively large and irregular space polygons. The links of horizontal transition spaces are created by extracting the centerline based on the Voronoi polygons for the corresponding areas. The door nodes, as repositioned in step 2, are connected to the links of horizontal transition spaces within the shortest distances. These connections can then be added to those links. Similarly, elevators and stairs located in horizontal transition spaces are also connected to links of horizontal transition spaces within the shortest distances.

- Step 4: The indoor node and link layers are created by combining all nodes and links from the previous three steps; unique IDs are set for each node and link. Nodes intersecting with each link are searched through a spatial join operation. Afterwards, the connectivity attribute is generated in the link layer to complete the creation of the indoor network model. This connectivity attribute, which is stored in the link layer, can be used in the graph database conversion process in the following section 3.3.

The network model can be created by floors through the proposed process. A model for a multi-story building can be configured by adding vertical links between building installations (elevators and stairs). The process of vertically linking a multi-layered model is introduced in section 3.2.5. The detailed operations of each step for the automatic indoor network model generation are described in the following sub-sections.



[Figure 3-5] The proposed process for creating horizontal network model

3.2.1 Definition of nodes and links in the network model

The targets of the indoor network model are room, corridor, stair, elevator, and door. These objects are the PWMD-related features defined in the proposed data model in section 2 and involved in the extracted indoor structure maps as detailed in section 3.1. Target features are abstracted with nodes and links according to their characteristics. Relations between each feature are captured in the network model. The basic concept is that spaces where horizontal and vertical transitions occur are converted to links, whereas general spaces for specific activities are converted to nodes. Additionally, the intersections of links can be added as nodes. A detailed configuration of the indoor network model based on the proposed data model is shown in [Table 3-3].

[Table 3-3] Configuration of the indoor network model

Target feature	Feature in the proposed data model*	Feature of the indoor structure map	Designed feature of the indoor network model
Corridor	*PWMDCorridor* (*TransitionSpace*)	Enclosed area surrounded by extracted wall pixels	Both node, link with ‘corridor’ as label attribute
Room	*PWMDRoom* (*GeneralSpace*)		Both node, link with ‘room’ as label attribute
Elevator	*PWMDElevator* (*TransitionSpace*)	Elevator pixels	Node with ‘elevator’ as label attribute
			Horizontal link with ‘elevator hall’ as label attribute
			Vertical link with ‘elevator shaft’ as label attribute
Stair	*PWMDStair* (*TransitionSpace*)	Stair pixels	Node with ‘stair’ as label attribute
			Horizontal link with ‘staircase’ as label attribute
			Vertical link with ‘stairway’ as label attribute
Door	*PWMDDoor* (*ConnectionBoundary*)	Door pixels	Node with ‘door’ as label attribute

* The contents in parentheses represent IndoorGML features mapped with features in the indoor network model

In the proposed data model, *PWMDRoom,* which is a subclass of general space, inherits navigable space. Navigable space is represented as a state feature with point geometry in dual space by Poincare duality (Lee, 2004). Therefore, a representative point for a general space is extracted and defined as a room node in the network model. The centroids of spaces intersecting with elevator and stair pixels serve as elevator and stair nodes, respectively. Although the elevator and the stair are mapped as transition spaces, they can also be regarded as general spaces to approach or stay for the transition. In other words, as the actual travel occurs vertically, those spaces can be abstracted like general spaces in horizontal aspects. Therefore, elevators and stairs are also converted to nodes like rooms with the 'label' attribute.

Spaces with large and complex shapes are likely to occur as horizontal transitions. Moreover, it is inadequate to abstract them as single nodes, because there may be factors that partially obstruct the movement of PWMD. In the proposed data model, these areas are defined as the *PWMDCorridor* feature, and it is necessary to increase the level of detail by using both nodes and links for corridor spaces. Also, corridors can be represented by centerlines (Goetz, 2012; Goetz & Zipf, 2011; Karas *et al.*, 2006; Teo & Cho, 2016). The centerline of the corridor is extracted and connected to associated door nodes within the shortest distance. Intersections, which are used to decide which path to be chosen, can be stored as corridor nodes. Corridor centerlines are split with those corridor nodes and stored as corridor links. Door nodes are created at the inner entrance of each space, and each indoor space can be connected to the corridor space through associated door nodes (Liu & Zlatanova, 2011). Vertical links for inter-floor transitions can be generated as connections between the pair of elevator and stair nodes on the lower and upper floors.

3.2.2 The classification rule of space polygons

According to Karas *et al.* (2006), the main backbone, which includes the connection between rooms and other indoor entities in the floor plan, should be considered a corridor. As mentioned earlier, to adequately compute costs, horizontal transition spaces such as corridors should be configured as separate sub-graphs, unlike general spaces represented by nodes (Karas *et al.*, 2006; Meijers *et al.*, 2005). Teo & Cho (2016) abstracted corridor spaces to links, as they considered corridors as building elements connected to other spaces. An area of sufficient size and complex shape with a high possibility of occurring horizontal transition requires a different procedure of abstraction from the general space.

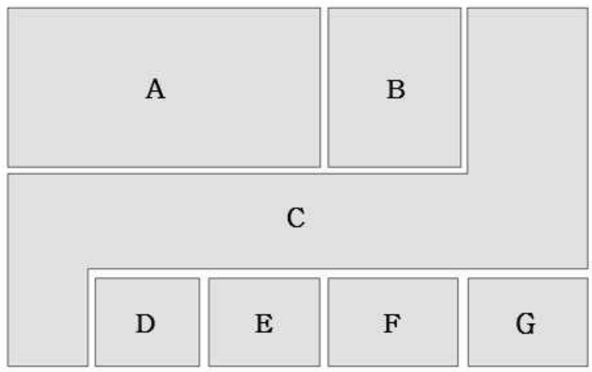
The previous method using source data that involve the space class as semantic information is advantageous to process corridors separately since corridors have already been defined among spaces. For example, corridors and standard rooms (general spaces) can be classified through the ifcRoom class in IFC files(Teo & Cho, 2016). However, the indoor structure map from section 3.1 does not contain space class and can only recognize spaces enclosed by walls. That is, the way to recognize corridors among enclosed spaces needs to be defined. Therefore, it is necessary to establish the rule for the detection of horizontal transition spaces.

In the proposed technique, the detection of horizontal transition spaces (which are represented as corridors hereinafter) is performed based on their area ratio(R) and shape complexity(S). Referring to the concept that small buildings are simpler than large buildings, space size is correlated with shape complexity (Burghardt & Steiniger, 2005). A space with a complex shape that occupies more than a certain proportion of the entire area incurs frequent movement by people within the room. Accordingly, spaces with high area ratio and shape complexity values (i.e. over given thresholds) are extracted as corridors. Spaces that are commonly recognized as corridors can be determined using an AND operation between these extracted spaces.

The R is calculated as the ratio of each space's area to the area of the entire floor. S is representative with the shape index as follows:

$$S = p/2\sqrt{\pi \cdot a} \tag{3-4}$$

Where p and a are the perimeter and area of each space polygon, respectively. As shown in [Figure 3-6], space C, which has a relatively high shape index, is also regarded as having a high shape complexity value, following Equation (3-4).



Space	Shape Index(S)
A	4.23
B	4.02
C	6.52
D	4.01
E	4.03
F	4.08
G	4.04

[Figure 3-6] Various space polygons with different shape index value

In summary, the rule for detecting horizontal transition space is defined as presented in [Table 3-4].

[Table 3-4] The rule for detecting horizontal transition space

Horizontal transition space detecting rule	
IF	[{Area ratio R > Threshold1}∧ {Shape complexity S > Threshold2}]
THEN	Horizontal transition space
ELSE	General space

The two thresholds (Threshold 1 for the area ratio, Threshold 2 for the shape complexity) used in [Table 3-4] are determined using a Precision-Recall (PR) curve (Davis & Goadrich, 2006). The PR curve is used to calculate the Equal Error Rate (EER)[6] when the data distribution is skewed. The precision(P) and recall(R) are calculated as follows:

$$\mathrm{P} = \frac{\mathrm{Count(detected\ corridor \cap actual\ corridor)}}{\mathrm{Count(detected\ corridor)}} \tag{3-5}$$

$$\mathrm{R} = \frac{\mathrm{Count(detected\ corridor \cap actual\ corridor)}}{\mathrm{Count(actual\ corridor)}} \tag{3-6}$$

As the thresholds change when applying the rule, the number of detected corridors varies, so the accuracy and recall also change. According to this change, a trade-off between precision and recall can be found; this point is selected as an EER. EER, the value with a balance of false acceptance rate and false rejection rate, can be used to select an optimal threshold (Bengio *et al.* 2005). For example, Kim (2014) used PR curves to select the optimal thresholds of shape similarity to search matched pairs between polygons as the distribution of matched class is skewed. Similarly, the distributions of two space

6) EER is the predetermined threshold values for false acceptance rate and false rejection rate.

classes- corridors and general spaces- are skewed in the binary classification of corridors and general spaces since the number of corridors is much smaller. Therefore, it is proper to apply the PR curve. In this study, PR curves for area ratio and shape complexity are plotted, respectively, to determine two thresholds for the rule in [Table 3-4]. The optimal thresholds of the area ratio and shape complexity are selected based on PR curves.

The overall procedure of space classification is shown in [Table 3-5]. Enclosed spaces are searched for with the wall raster map, and bounding boxes of extracted elevator and stair pixels are polygonized. Then, corridors and rooms are classified according to the rule presented in [Table 3-4], and elevator and stair polygons located on the corridors are determined through overlay analysis. These elevator and stair polygons are processed in a later step along with corridor spaces.

Spaces other than corridors can be abstracted as centroid-based nodes in the indoor network model (Goetz, 2012; Karas *et al.*, 2006; Teo & Cho, 2016; Yang & Worboys, 2015). Therefore, the centroids of the remaining polygons can be extracted as general space nodes after corridor selection. The nodes of elevators and stairs can be distinguished from general space nodes using the ‘label’ attribute.

[Table 3-5] The proposed procedure for generating nodes of general spaces with the label attribute

Procedure 1	**Node generation of general spaces with label attribute**
Input	Extracted wall, elevator, stair raster layer
Output	General space node layer with label attribute
Intermediate*	Elevator and stair polygon layer Polygon layer of elevators and stairs within corridor Horizontal transition space(Corridor) polygon layer General space polygon layer with label attribute

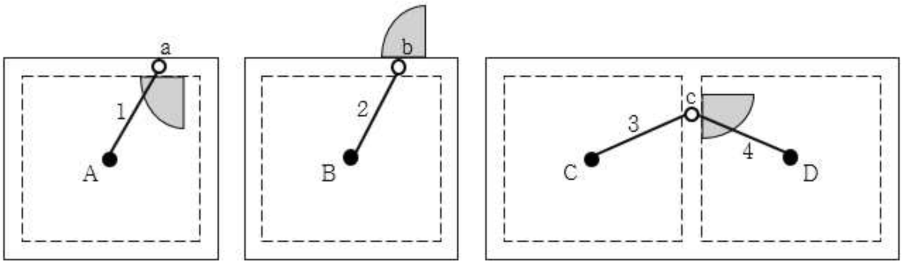
1 **read** Wall raster layer
2 **create** spaces enclosed with wall pixels as space polygons ($S=\{s_1,\cdots,s_n\}$)
3 **create** bounding boxes of stair, elevator pixels as elevator and stair polygons ($P_e=\{p_{e_1},p_{e_2},\cdots,p_{e_k}\}$, $P_s=\{p_{s_1},p_{s_2},\cdots,p_{s_m}\}$)
4 **assign** id to P_e, P_s
5 **return** Elevator and stair polygon layer
6 **for** n **in** S
  1 **if** $R(s_n)$ > threshold 1 **and** $S(s_n)$ > threshold 2 :
    1 **append** s_n to Corridor polygon layer
7 **return** Horizontal transition space(Corridor) polygon layer
8 **for** k **in** P_e (iteration for P_s)
  1 **if** centroid(p_{e_k}) is within Corridor polygon(P_c)
    1 **append** p_{e_k} **to** Polygon layer of elevators (stairs) within corridor
9 **remove** P_c from S as general space polygons(P_g)
10 **extract** P_g **intersecting** centroid(P_e, P_s)
11 **assign** ‘stair’, ‘elevator’ respectively to ‘label’ attribute
12 **for** t in P_g
  1 **if** ‘label’ is **Null**:
    1 **assign** ‘room’ to ‘label’ attribute
13 **return** General space polygon layer with ‘label’ attribute
14 **extract** centroid(P_g) as general space nodes with ‘label’ ($N_g=\{n_{g_1},\ldots,n_{g_t}\}$)
15 **return** General space node layer with ‘label’ attribute

* temporary output for latter process

3.2.3 Connection between general spaces and doors

Each indoor space is accessed through internal entrances. Therefore, room nodes must be connected to corridors through associated door nodes (Teo & Cho, 2016; Yang & Worboys, 2015). IFC files store the relations of indoor entities as attributes such as IfcRelSpaceBoundary, whereas indoor structure maps contain insufficient information to select the associated room for each door. Therefore, after determining which spaces to connect or disconnect each door to, connections from doors to associated rooms can be created. In other words, a link is created after retrieving associations based on the spatial relationship between the extracted door and the room in the proposed procedure.

Types of the relationship between the adjusted doors and spaces which are enclosed by walls are depicted in [Figure 3-7] according to the new annotations described in section 3.1.2. In [Figure 3-7], the door is located inside the room enclosed by walls, located inside the corridor while outside the room, and located between two adjacent rooms. In consideration of these relationships, each room's associated doors can be searched through a spatial join between buffered room polygons with the wall thickness and the adjusted door node. Since the connection link is created after adjusting door positions by snapping to the wall line, a 'one-to-many join' occurs when a door is located between adjacent rooms. In this case, several links connecting a door with multiple rooms are generated (The door 'c' in [Figure 3-7]).



Connection	Join door id (start node)	Target room id (end node)
1	a	A
2	b	B
3	c	C
4	c	D

Room buffer polygon Room polygon Room node
Extracted door pixel cluster Adjusted door node
Room to door link (connection)

[Figure 3-7] Types of spatial relationships between rooms and associated doors

Room nodes are connected to the associated doors through the proposed procedure([Table 3-6]) by applying the approach shown in [Figure 3-7].

[Table 3-6] The proposed procedure for connecting general spaces and associated doors

Procedure 2	**Connection between general space and door**
Input	Extracted door, wall raster layer, General space polygon/node layers with label attribute
Output	Adjusted door node layer Room to door link layer

Step1: Create door node layer

1 **convert** wall raster into wall polyline using skeletonization

2 **for** n **in** all extracted doors ($P_d = \{p_{d_1}, p_{d_2}, \cdots, p_{d_n}\}$)

  1 **extract** centroid(p_{d_n})

  2 **snap** centroid(p_{d_n}) to nearest wall line($l'_{wall} \in l_{wall}$) as n_{d_n}

  3 **if** $\| n_{d_n} - n_{d_{nearest}} \| <$ door width:

    1 **replace** ($n_{d_n}, n_{d_{nearest}}$) to $(n_{d_n} + n_{d_{nearest}})/2$ as n'_{d_n}

    2 **remove** ($p_{d_{nearest}}$) from P_d

  4 **append** n'_{d_n} to Adjusted door nodes ($N_d = \{n_{d_1}, n_{d_2}, \cdots, n_{d_m}\}$)

3 **return** Adjusted door node layer

Step2: Create Room to door link layer

1 **create buffer** P_g with wall pixel distance

2 **join** N_d with **intersecting** P_g based on **'one-to-many'** type

3 **copy** joined room_id(P_g) to N_d as 'connected_room_id'

4 **for** m **in** N_d

  1 **search** n'_g with identical id in N_g to 'connected_room_id'

  2 **connect** lines between n'_g and n_{d_m}

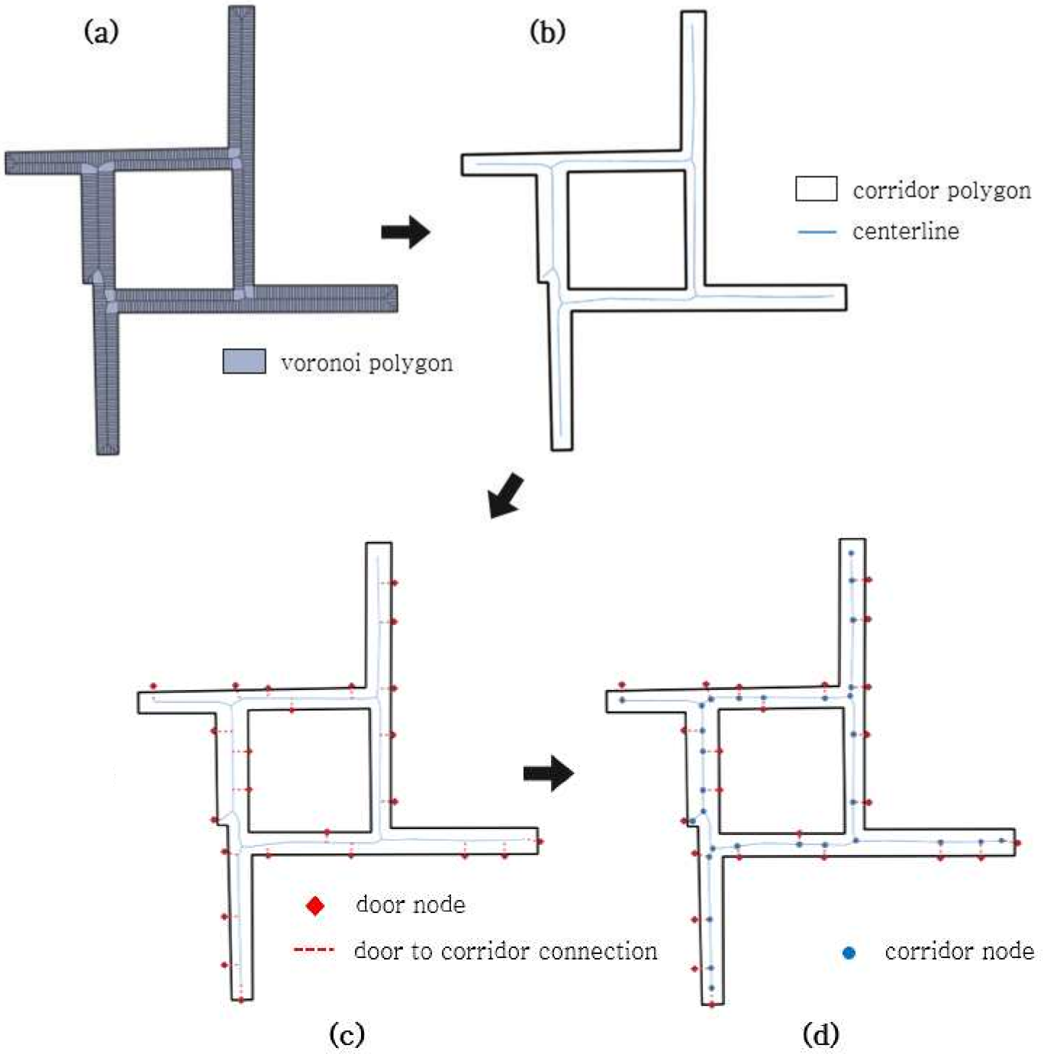
5 **return** Room to door link layer

3.2.4 Node-link generation for horizontal transition spaces

Automatically creating horizontal routes using corridor boundaries is challenging (Teo & Cho, 2016). In many previous studies, centerlines based on Voronoi diagrams or medial axes have been used to model the corridor space (Goetz & Zipf, 2011; Karas *et al.*, 2006; Lee, 2004; Taneja *et al.*, 2011; Teo & Cho, 2016; Yang & Worboys, 2015). Although their algorithms have been refined sequentially, the basic principles of these algorithms are identical (Taneja *et al.*, 2016). As people tend to move in the middle of a corridor, rather than close to the wall, modeling a corridor with a centerline can properly represent human behavior within indoor environments (Goetz & Zipf, 2011; Mortari *et al.*, 2019).

The centerlines of corridors can be extracted as corridor links using the detected boundaries of horizontal transition space polygons. Since the corridor polygons are formed over the area surrounded by wall pixels, the corridor boundaries are not neat. Thus, the input corridor is unsuitable for applying the MAT-based methods that extract centerlines based on the angle bisector at each vertex of the corridor boundary. Also, it is difficult to perform precise computation using the boundary. Therefore, a procedure for abstracting corridors suitable for target input is proposed by modifying previous centerline extraction method based on Voronoi diagrams in Dilts (2015). As each Voronoi region is closest to the vertices on the boundary that encloses it, the resulting skeleton is equidistant from at least two vertices on the polygon boundary. Therefore, a centerline can be generated for any planar polygon (Taneja *et al.*, 2016). Based on this principle, the idea of the proposed procedure is that the centerline is created by extracting the Voronoi polygons' boundary that is included completely in the corridor space ([Figure 3-8(a), (b)]). Corridor space boundaries are converted into points with regular intervals, and each point is used as seed points to create Voronoi polygons ([Figure 3-8(a)]).

Additionally, the shortest path from each door to the corridor centerline can be added to corridor links ([Figure 3-8(c)]). Door nodes can be perpendicularly projected onto the corridor route to connect doors to the corridor (Teo & Cho, 2016). If perpendicular points cannot be generated on the corridor links, then they link to the nearest point on the corridor. Moreover, considering elevators and stairs located in corridors, rather than enclosed rooms, the corridor link creation is completed by adding the shortest connection from those facilities to the corridor links. In particular, the stair can exists in the closed room or open spaces such as corridors. In the proposed procedure, the connection detail is improved by processing the two types of stairs separately. Finally, the intersections and endpoints of corridor links are stored as corridor nodes ([Figure 3-8(d)]).



[Figure 3-8] Sequential process of generating corridor links and nodes

The generation procedure for corridor node and link layers is described in detail in [Table 3-7].

[Table 3-7] The proposed procedure for generating nodes and links of horizontal transition space

Procedure 3	Node, link generation of horizontal transition spaces
Input	Horizontal transition space(Corridor) polygon layer Polygon layers of elevator and stair within corridor Adjusted door node layer General space node layer with label attribute
Return	Horizontal transition space node/link layers General space node layer with label attribute (updated)
Intermediate	Temporary corridor link layer

Step1: Centerline extraction

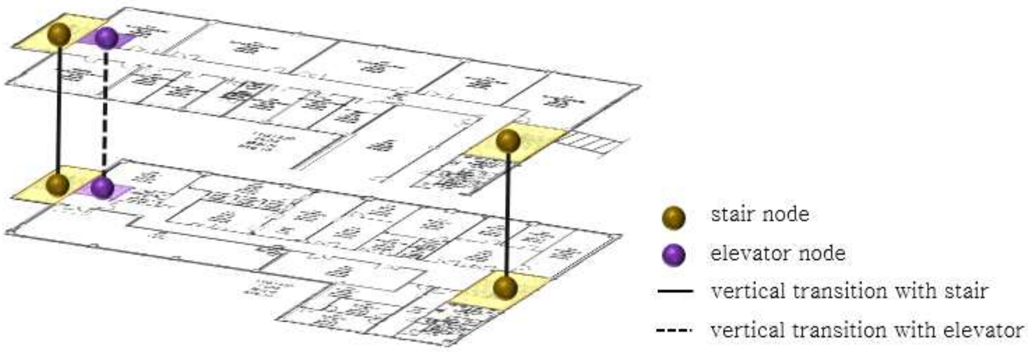
1 **read** Horizontal transition space polygon layer
2 **for** n **in** P_c
  1 **extract** boundary of the p_{c_n}
  2 **add** vertices($V=\{v_1,v_2,\cdots,v_k\}$) along boundary
  3 **create** Voronoi polygons($P_V=\{p_{v_1},p_{v_2},\cdots,p_{v_k}\}$) with each vertex as a seed point
  4 **detect** edges of P_V which are **completely within** p_{c_n}
  5 **dissolve** detected edges as corridor links(l_{c_n})
3 **return** Temporary corridor link layer(l_c)

Step2: Create corridor link layer

1 **read** Adjusted door node layer, Corridor polygon layer
2 **create buffer** layer of P_c with wall pixel distance
3 **detect** N_d **intersecting** buffered(P_c) as N_d'
4 **read** Temporary corridor link layer
5 **for** m **in** $N_d'=\{n'_{d_1},n'_{d_2},\cdots,n'_{d_m}\}$
6 1 **add** shortest line from n'_{d_n} to l_c
7 **read** Polygon layers of elevator and stair within corridor
8 **for** t **in** P_e (iteration for P_s)
  1 **extract** centroid(p_{e_t})
  2 **append** centroid(p_{e_t}) to N_g with 'elevator' ('stair') label
  3 **add** shortest line from centroid(p_{e_t}) to l_c
9 **return** General space node layer with label attribute
10 **create** nodes at intersection of l_c as corridor node layer(N_c)
11 **split** l_c with N_c
12 **add** end points of l_c to N_c
13 **return** Horizontal transition space node/link layer

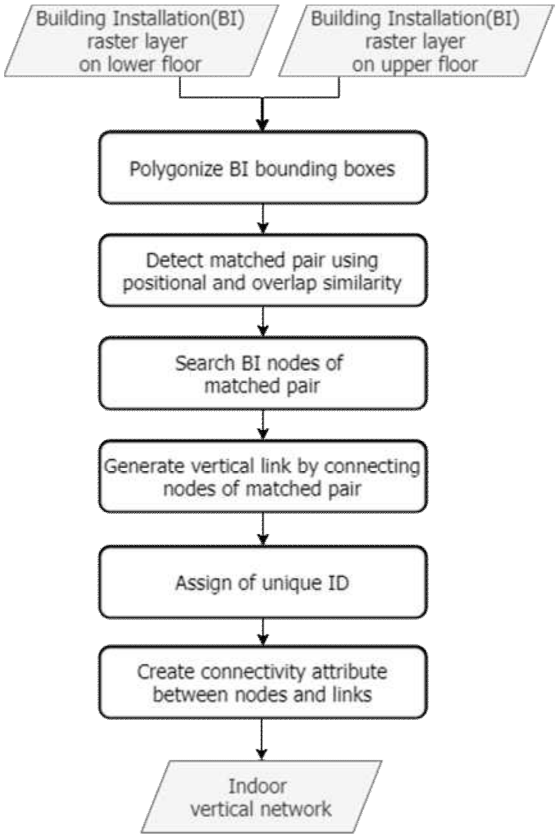
3.2.5 Vertical link generation

The procedure for generating vertical links was developed by modifying the workflow presented in Park & Yu (2019). The vertical connectivity relations among floors are defined using stairs and elevators (Lee, 2004). Building installations for vertical transitions such as elevators and stairs are illustrated at identical positions on each floor in the floor plans. Therefore, the vertical links can be created by connecting successive elevators and stairs on the upper and lower floors ([Figure 3-9]).



[Figure 3-9] The concept of generating vertical connections

Many elevators and stairs exist in complex buildings and scanned floor plans are not precisely aligned by floor. Therefore, features regarded as successive pairs must be detected and then connected. This study proposes an algorithm that automatically generates vertical links for inter-floor transitions by connecting facilities such as elevators and stairs ([Figure 3-10]).



[Figure 3-10] The proposed process for generating vertical links

Although scanned floor plans are not accurately aligned, the scale and the extent of the pixel coordinates of all floor plans for one building are identical. Furthermore, pairs of elevators and stairs on successive floors are illustrated at the same position in the floor plans. Under these characteristics, matched-pairs that need to be connected are determined using positional and overlap similarities (Kim *et al.*, 2011). The positional similarity(P_{ij}') and overlap similarity(A_{ij}') can be calculated as shown in Equations (3-7) and (3-8). Optimal matched-pairs of successive facilities can be identified through a linear combination with identical weights of those two similarities.

$$P_{ij}' = 1 - \frac{P_{ij}}{\max(P_{ij})}$$

$$where \;\; P_{ij} = \sqrt{(X_i - X_j)^2 + (Y_i - Y_j)^2}, \tag{3-7}$$

$$(X_n, Y_n) = \;\; Centroid \;\; coordinate \;\; of \;\; polygon \;\; n$$

$$A_{ij}' = 1 - \frac{A_{ij}}{\max(A_{ij})}$$

$$where \;\;\; A_{ij} = \left| \frac{A_{A \cup B} - A_{A \cap B}}{A_i + A_j} \right| \tag{3-8}$$

$$A_n = The \;\; area \;\; of \;\; polygon \;\; n$$

When the optimal matched-pair is determined, the 'id' of the facility feature on the upper floor is stored as an attribute of its counterpart on the lower floor. A vertical link can be created by connecting the matched facility nodes of consecutive floors based on stored 'id'. [Table 3-8] presents the generation procedure of a vertical link for an elevator shaft with pseudo-code. The link for the stairway can also be created by applying an identical process to the stair raster map.

[Table 3-8] The proposed procedure for generating vertical links

Procedure 4			**Vertical link generation**
Input			Elevator polygon layer
Return			Vertical links of elevators
1			**read** Elevator polygon layer
2			**for** i **in** elevator features on lower floor ($P_{e_i}^{lower}$)
	1		**for** j **in** elevator features on upper floor ($P_{e_j}^{upper}$)
		1	**calculate** A_{ij} and P_{ij}
3			**extract** $\max(A_{ij})$ and $\max(P_{ij})$
4			**for** i **in** $P_{e_i}^{lower}$
	1		**for** j **in** $P_{e_j}^{upper}$
		1	**calculate** A_{ij}' and P_{ij}'
		2	**generate** $M[i][j] = 0.5 \times A_{ij}' + 0.5 \times P_{ij}'$
5			**for** i **in** elevator features on lower floor
	1		j = M.argmax(axis=1)[i]
	2		**search** node of P_{e_i} and P_{e_j}
	3		**connect** node of $P_{e_i}^{lower}$ with node of $P_{e_j}^{upper}$
	4		**save** connection line as vertical link of elevator
6			**return** Vertical links of elevators

3.2.6 Connectivity and accessibility information generation

The procedures proposed in sections 3.2.2–3.2.5 can create three node layers (room, door, and corridor) and two link layers (room to door and corridor). After combining the node layers and link layers separately, 'node id' and 'link id' are set as unique identifiers.

A link segment consists of two nodes: a start node and an end node. The ids of two nodes spatially joined to each link can be stored as a 'start node id' and an 'end node id'. As two-way travel is possible within indoor environments (except for within escalators), these two nodes can be regarded as both start and end. Therefore, the order in which the nodes are stored in the link layer is irrelevant. Additionally, a few bent corridor links caused during centerline creation can be adjusted to straight lines through reconnection between start and end nodes according to the connectivity information.

The accessibility values of indoor features are calculated using proposed indices in 2.3.2. Accessibility is measured for each component, and the calculated values are stored in node layers. The total accessibility of each indoor feature can be calculated by summing up the indices of its individual factors. Link accessibility is assigned by selecting the minimum values from the calculated accessibilities of both connected nodes. The calculated accessibility value can be converted into a binary value (accessible or inaccessible) by comparing it with the thresholds. The binary value of links are determined as True if either of the two connected nodes is inaccessible (OR operation). It can then be stored as a 'inaccessible' property of an indoor graph database. This 'inaccessible' property can be used to extract feasible networks for PWMD routing.

3.3 Indoor graph database for PWMD

3.3.1 Graph representation of indoor environments

This section describes the process of graph modeling to convert the conceptual data model presented in section 2 into a graph model. Graph models are very useful for expressing complex relationships among building elements, and for analyzing topologies (Ismail *et al.*, 2017; Khalili & Chua, 2015).

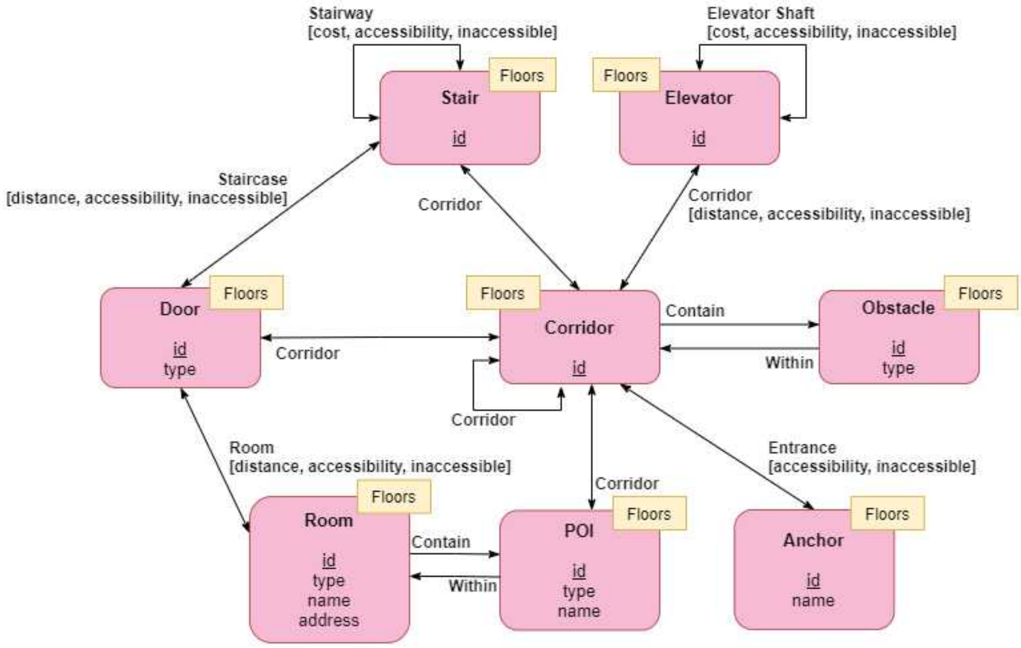
Roy-Hubara *et al.* (2017) and De Sousa & Cura (2018) performed a logical design of a graph database based on the Entity-Relationship (ER) conceptual model. The general process of graph-based schema design was proposed in these studies. The labels and properties of the edges and vertices of the graph model can be designed through mapping with the ER model, and the constraints can also be defined. Graph modeling has also been conducted using a specific target source. Steinmetz *et al.* (2018) used OSM's node and way elements as sources and converted them into nodes and relationships in the Neo4j graph database. The node and way elements of OSM can correspond to the nodes and links of the created indoor network model. In this study, by referring to the modeling in previous works, the node and link layers of the created network model are converted into the nodes and relationships of the graph database, respectively. The indoor features are expressed by using a named graph model, and the stored attributes of the network model are converted into the properties of named graphs.

The graph database model can be defined as follows (De Sousa & Cura, 2018):

$$G = (LV, LE, PV, PE, CV, CE) \tag{3-9}$$

Where LV, LE represent a set of labels on vertices and edges, respectively; PV, PE represent a set of properties on vertices and edges, respectively; CV, CE represent a set of constraints on vertices and edges, respectively. The constraints (CV, CE) define that the labeled vertices and edges require properties, including the identifier. In summary, each labeled vertex must be linked to another vertex with an edge under constraints. However, cardinality constraints are not considered in the proposed graph model, as they have not been defined yet in Neo4j (Roy-Hubara *et al.*, 2017).

[Figure 3-11] shows the graph model created based on the conceptual data model presented in section 2. All nodes are classified by floor using the 'Floors' label. Each node represents a space such as a room, stair, or elevator. The named relationship describes horizontal connections between these spaces or vertical connections through stairs or elevators. Rooms and corridors are usually connected through a door, and if there is a door between two adjacent rooms, those rooms are connected through said door. Therefore, *Room* and *Corridor* features should refer to a *Door* feature by relationships named *Room* and *Corridor,* respectively. Two-way travel is feasible within indoor environments, so bi-directed graphs are used. Hor *et al.* (2018) and Ismail *et al.* (2018) represented corridors as single nodes, as they expressed all indoor spaces as nodes. As discussed earlier, unlike an area such as a simple small room, both accessible and inaccessible spots can be included in a single corridor because of the shape's complexity and size. This characteristic of corridors is crucial regarding PWMD indoor navigation, and so must be reflected. Thus, corridors are represented as sub-graphs in the proposed indoor network model considering routing applications. Therefore, as shown in [Figure 3-11], corridor features are modeled as both nodes and relationships including recurrent pattern.



[Figure 3-11] Graph model based on the proposed data model ([Figure 2-3])

The indoor network model comprises a relational database storing the geometries and attributes in the table. If all attributes are to be stored as node properties in the converted graph database, then the database would become heavy. Furthermore, its efficiency may decrease when executing queries such as path calculations. Therefore, instead of storing all attributes in the graph database, accessibility values calculated using the designed index are defined as the 'accessibility' property of each relationship. The binary 'inaccessible' value is stored as a property of relationships in the graph model to query feasible graphs for PWMD routing.

3.3.2 Conversion of network model into graph database

A graph database is a powerful tool for managing data in terms of relationships between entities (Hor *et al.*, 2018). A graph database is a set of property graph models, and a property graph consists of nodes with various attributes as key-value pairs. Nodes are semantically and meaningfully connected through relationships, and relationships have direction, type, and start-end nodes (Ismail *et al.*, 2018). In this context, the indoor network model proposed here is converted into the Neo4j graph database.

Spatial data such as shapefiles can be imported using the Neo4j spatial library[7]. However, all features in the spatial layer are imported only as nodes in the graph database. In other words, there is a limitation in that geometry types (point, polyline, or polygon) cannot be reflected when spatial data is converted to a graph database using the import function of the Neo4j spatial library. Specifically, as a result of the modeling presented in section 3.3.1, the link layer, which represents relations between nodes, should be converted to corresponding labeled relationships in the graph database. However, this process cannot be achieved with an existing library. Accordingly, this study proposes a method for converting the indoor network model to the graph database, based on graph modeling in section 3.3.1. Contrary to previous researches (Hor *et al.*, 2018; Ismail *et al.*, 2018), if spaces, which are inappropriate to be represented with a single node, are abstracted as sub-networks with both nodes and links, the network model is converted to form sub-graphs in the graph database also. Moreover, the network model can be converted to configure a multi-layer graph,

7) Neo4j Spatial is a utility library for Neo4j that enables spatial operations on data. https://neo4j-contrib.github.io/spatial/

including both horizontal and vertical connections. That is, the characteristics of the network model can be preserved while converting into a graph database in the proposed technique. This conversion can be executed using Cypher commands. In the proposed technique, all of the indoor nodes can be imported into Neo4j with labels, and named relationships can be created using the link layer. The automatically written Cypher commands include the property conversion for all nodes and relationships.

In this study, the proposed technique for converting from an indoor network model to a Neo4j graph database is performed in two steps:

1) Import the node layer of the indoor network model by floor, and create multiple node labels for the number of floors and the relevant building's name.

2) Create named relationships between imported nodes based on the derived graph pattern.

The node layer of the indoor network model can be imported into Neo4j using relevant libraries. Then, the nodes' labels can be set using the building name and floor number. All Cypher commands necessary to create named relationships in Neo4j based on the link layer of the indoor network model are written automatically. [Table 3-9] shows an example of Cypher commands to import nodes with label.

[Table 3-9] The example of Cypher commands for importing nodes with label

```
# import node layer of F1
CALL spatial.importShapefile ("node_F1.shp")
# set label of floor number(F1) to nodes on F1
MATCH (n) SET n:F1
# set multi-label of floor number(F1) and the building name to nodes on F1
MATCH (n) SET n:B222:F1

# import node layer of F2
CALL spatial.importShapefile ("node_F2.shp")
# search nodes without label 'F1' and set label of 'F2'
MATCH (n) WHERE size(labels(n))=0 SET n:F2
```

In the next step, direct relationships are created based on the connectivity information of the imported nodes. [Table 3-10] presents pseudo-code for the proposed procedure of creating named relationships between imported nodes, in accordance with the patterns derived from graph modeling.

The 'MATCH' and 'CREATE' query forms used in the procedure in [Table 3-10] are as follows.

- MATCH (n1:{nodeLabel}) WHERE n1.id={startNodeId}
- MATCH (n2:{nodeLabe2}) WHERE n2.id={endNodeId}
- CREATE (n1)-[r1:{relationshipType} {cost: {dist/acValue}, inaccessible: {inacc}, distance: {dist}]->(n2)

[Table 3-10] The proposed procedure for creating named relationships

Procedure 5	**Automatic Cypher commands generation**
Input	Link layer of indoor network model
Return	Cypher commands

1 **read** link shapefile
2 **for** n **in** all link features $l \in L$
  1 start(n) = ‘start node id’ from link feature(n) attributes
  2 end(n) = ‘end node id’ from link feature(n) attributes
  3 relation(n) = ‘label’ from link feature(n) attributes
  4 inaccessible(n) = ‘inaccessible’ from link feature(n) attributes
  5 accessibility(n) = get ‘accessibility’ from link feature(n) attributes
  6 floors(n) = ‘floors’ from link feature(n) attributes
  7 **if** (vertical):
    1 lower(n) = get ‘upper floor’ from link feature(n) attributes
    2 upper(n) = get ‘lower floor’ from link feature(n) attributes
    3 distance(n) = calculate vertical length of link feature(n)
  8 **else:**
    1 lower(n) = floors(n)
    2 upper(n) = floors(n)
    3 distance = calculate length of link feature(n)
  9 **for** n **in** link features:
    1 **get** ‘MATCH’ query form
    2 **input** lower(n) to ‘nodeLabel1’, and upper(n) to ‘nodeLabel2’
    3 **input** start(n) to ‘startNodeId’ and end(n) to ‘endNodeId’
    4 **get** ‘CREATE’ query form
    5 **input** relation(n) to ‘relationshipType’, distance(n) to ‘dist’, accessibility(n) to ‘acValue’, and inaccessible(n) to ‘inacc’
  10 **copy** ‘CREATE’ commands to swap (node 1) and (node 2)
  11 **save** generated Cypher commands
3 **return** Cypher commands list

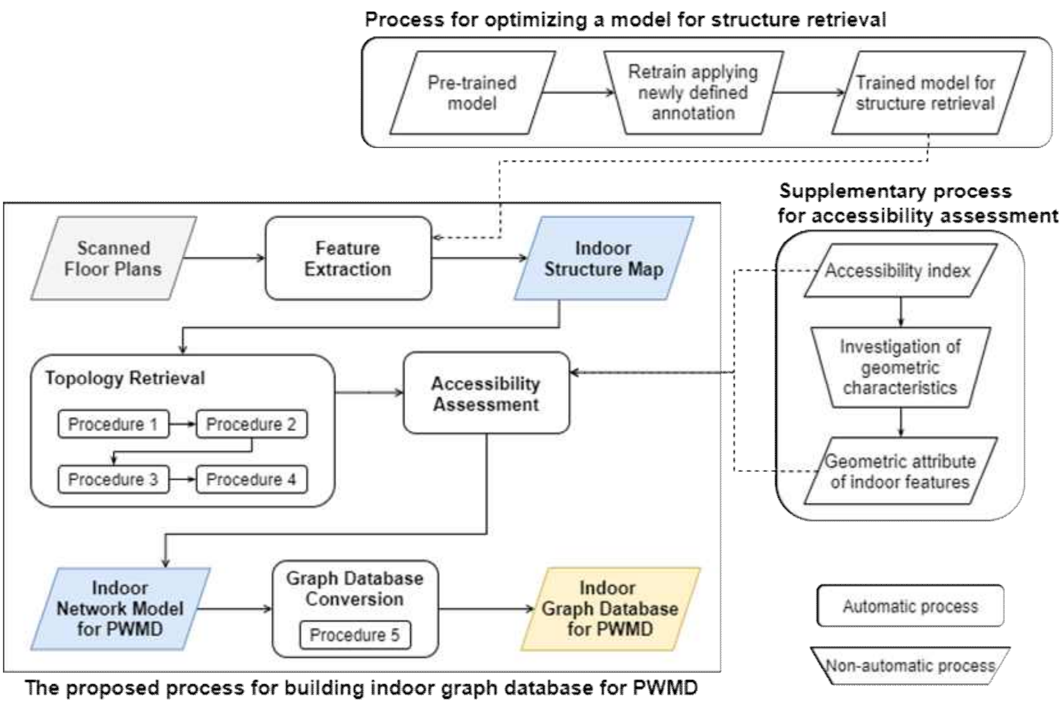
3.4 Entire process

[Figure 3-12] presents a proposed technique of generating an indoor graph database for PWMD in this study, including each step-by-step process described in 3.1-3.3. Using the proposed process represented in the box in [Figure 3-12], the indoor graph database for PWMD is generated as follow: When a scanned floor plan is inputted, an indoor structure map is generated through feature extraction using the trained model. Then, an indoor network model for PWMD can be created through accessibility assessment after topology retrieval using the generated indoor structure map. The final indoor graph database for PWMD can be generated by converting the network model into a graph database. In the step of topology retrieval using the indoor structure map, four automated sub-procedures (Procedures 1-4 in 3.2) are included, and for the graph database conversion, a sub-procedure (Procedure 5 in 3.3) is included. Compared with previous methods in generating navigation databases, the proposed technique is differentiated in that the sub-procedures are organically linked as one process to generate the target database from scanned floor plans while retrieving information such as the association between indoor features lost in scanned floor plans.

The trained model for indoor feature extraction is prepared by retraining the pre-trained model while applying the changed annotation considering topology retrieval. Depending on floor plans to be input, forms of expressing indoor objects might be different; thus, the model for feature extraction needs to be fine-tunned when applying the proposed process.

Multiple sets of floor plans have similar primary objects (walls, doors, etc.) commonly expressed. Although the expressing method might be varied, many rule-based detecting methods for floor plan vectorization based on the relatively standardized symbols have also been developed. That is, there is common knowledge about objects, even if there are sets of floor plans representing indoor objects in different ways. Therefore, it is advantageous to train a deep network model by sharing that common knowledge. Effective retraining is possible by applying the transfer learning-based approach of this study.

Investigation of detailed geometric factors for accessibility assessment such as width and the slope is conducted using supplementary documents or field surveys. Assuming that the investigation and model optimizing is completed as a separate process, the entire process, from inputting scan floor plans to generating an indoor graph database for PWMD, can be performed automatically.



[Figure 3-12] The proposed technique of generating an indoor graph database for PWMD

4. Experiment and results

4.1 Experimental setup and test data

An indoor graph database of Seoul National University was generated by applying the proposed method. The newly generated dataset includes four classes, excluding windows. The size of scanned floor plans in the dataset ranging from 1,271*1,271 to 6,034*6,034. For the training, Adam was used as the optimizer and LeakyReLU was used for the activation function. Also, the learning rate and style ratio were set as 0.00001 and 0.003, respectively. The classifier was retrained with 40 epochs, and all layers were fine-tuned with 30 epochs. Test floor plans were input as patches of 512*512 with 30% overlap as identical as pre-training. The transfer learning was implemented using Tensorflow 1.13. The process of generating the network model using the indoor structure map was implemented based on the graphical modeler of Qgis 3.10 and Python 3.7. To convert the network model into a graph database, a script for an Cypher command writer was developed using Python 3.7. Through the execution of the script, Cypher commands were written to appropriately convert all nodes and links of the indoor network model into a graph database. The indoor graph database was generated using Neo4j graph database version 3.5.19, and a routing test for validation was performed using Neo4j desktop 1.4.1.

A test set with seven different floor plans was selected to generate indoor graph databases applying the proposed technique. Kim (2020) attempted to verify the flexibility of the methodology by composing a test set including 'regular/irregular-shaped building' and 'building with complex features'. In this study, the criteria from Kim (2020) were specified, and buildings with various structures were configured in consideration of 'shape of floor layout', 'wall geometry', and 'type of transition spaces'. Criteria for the 'shape of floor layout' and 'wall geometry' were used to express the diversity of each building's boundary shape and each individual space's shape, respectively. 'Type of transition spaces' presents the various types of corridors. A detailed description of the test dataset is provided in [Table 4-1].

[Table 4-1] The detailed description of the test dataset

Test set	Shape of floor layout	Wall geometry	Type of transition spaces
A	Consistent, rectangle-shaped floor layout	Composed of regular straight walls	Including a narrow and long corridor
B	Consistent, relatively square-shaped floor layout	Composed of regular straight walls, including curved walls for the elevator shaft	Including a corridor surrounding rooms (polygon with hole)
C	Inconsistent floor layout, including irregular-shaped rooms	Including diagonal walls	Including an irregular-shaped corridor composed of diagonal walls
D	Floor layout in which two rectangles are orthogonal	Composed of regular straight walls	Including a narrow and long corridor and a corridor in the form of a wide open-space; Including two corridors separated by a door

Test set	Shape of floor layout	Wall geometry	Type of transition spaces
E	Irregular polygonal floor layout	Composed of long, oblique walls	Including a narrow and long corridor composed of oblique walls
F	Atypical floor layout with regular square-shaped individual rooms	Including relatively short curved walls	Including a corridor surrounding rooms (polygon with hole)
G	Floor layout in which two rectangles are orthogonal	Composed of regular straight walls	Including a corridor in which two narrow and long rectangles are orthogonal

To verify the appropriateness of the vertical link generation, multi-floor data were created for set-G by applying the proposed technique to scanned floor plans from the first to the fifth floors. For the constructed multi-floor set, a quality assessment of the construction result was performed through a routing test, including inter-floor transitions.

4.2 Evaluation for retrieved information

4.2.1 Results of structure retrieval

As the first stage, the proposed technique was applied on the test set to create indoor structure maps. A quantitative evaluation of the indoor structure map was performed using precision and recall. The TRUE detection was determined through a pixel-wise comparison of the classes (wall, door, elevator, and stair) between the annotated floor plan and the created indoor structure map. Specifically, precision was calculated as the proportion of extracted pixels as target features that match the ground truth among total extracted pixels. Recall was defined as the proportion of extracted pixels as target feature among pixels of target feature in ground truth. [Table 4-2] shows the quantitative and qualitative evaluation results for the indoor structure map for the test sets. The pre-trained model showed an average *F-measure* of 0.87 (precision of 0.89 and recall of 0.86) for the 30 test floor plans of the SNU dataset (Ministry of Land, Infrastructure, and Transport, 2020). The retrained model in this study has an average *F-measure* of 0.87 for 7 test sets, which shows a similar level of accuracy to the pre-trained model.

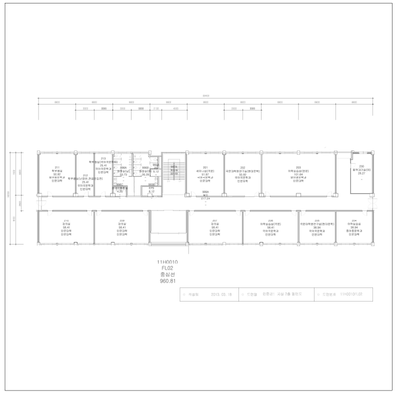
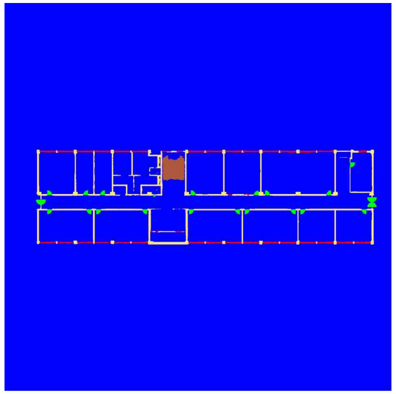
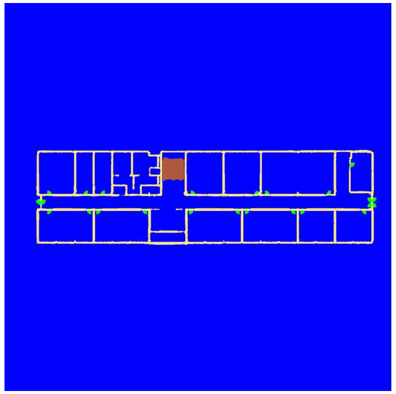
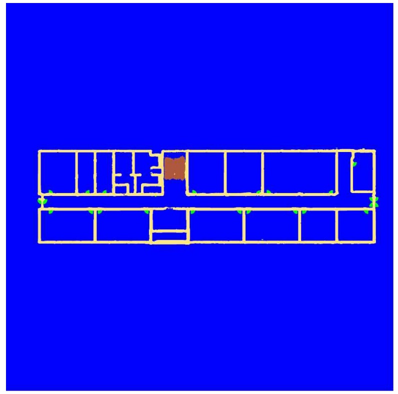
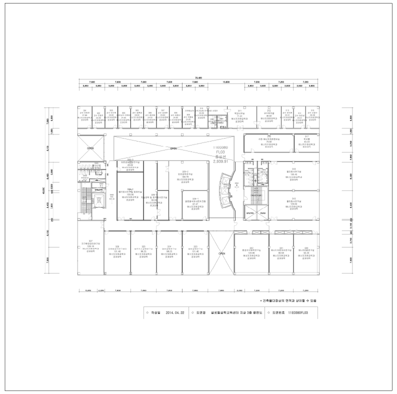
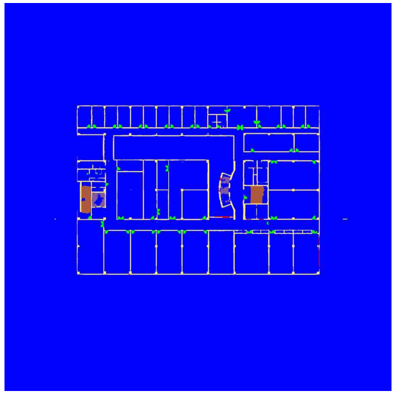
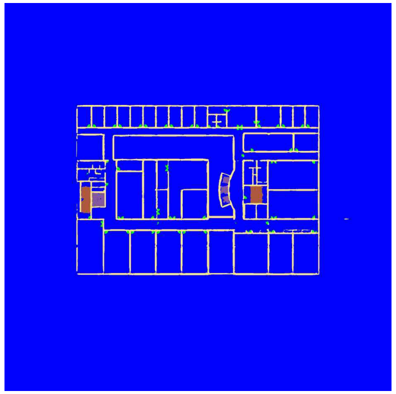
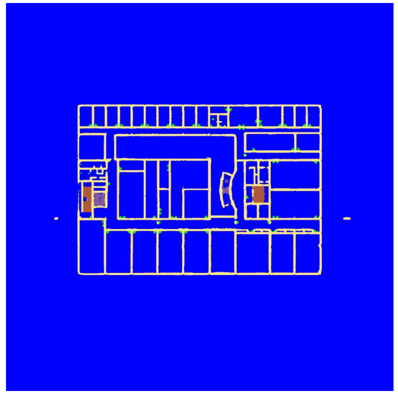
[Table 4-2] The evaluation results for indoor structure maps

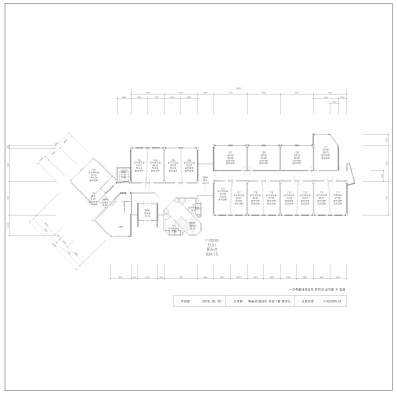
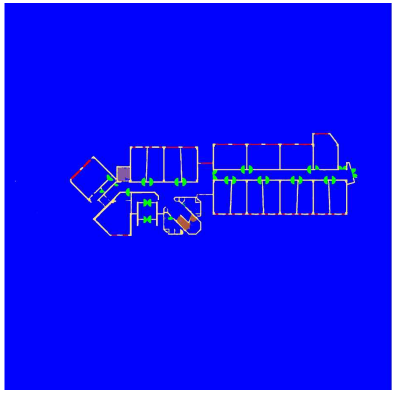
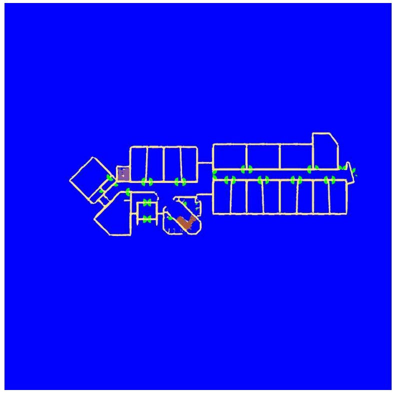
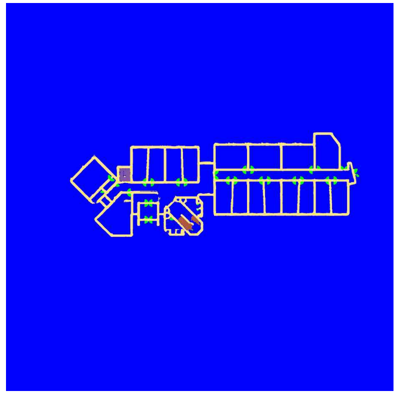
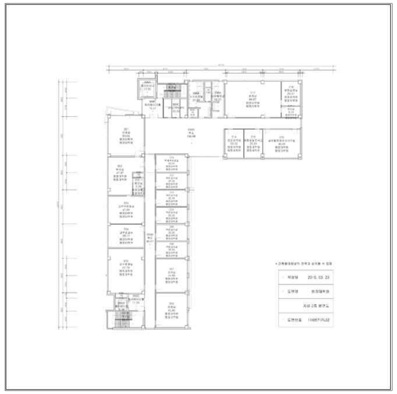
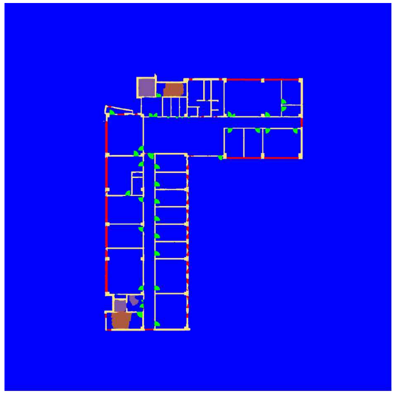
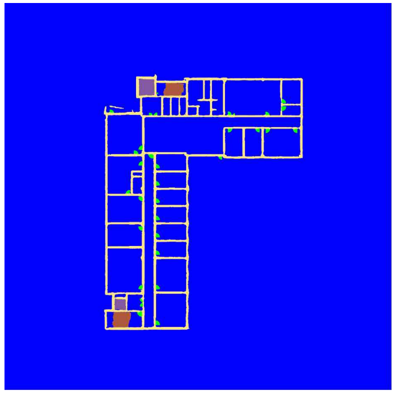
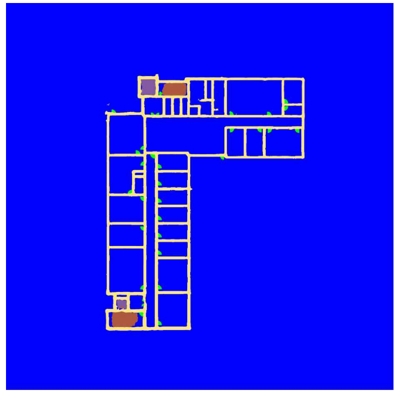
Test set	Quantitative evaluation		Qualitative evaluation	
	Precision	Recall	Reflection of changed annotation	The number spaces of unenclosed/total
A	0.90	0.92	both	1/24
B	0.90	0.81	both	4/68
C	0.81	0.88	both	2/32
D	0.91	0.83	both	0/36
E	0.87	0.86	both	5/32
F	0.87	0.90	both	0/32
G	0.84	0.89	both	0/53

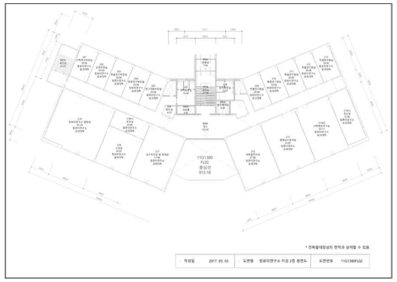
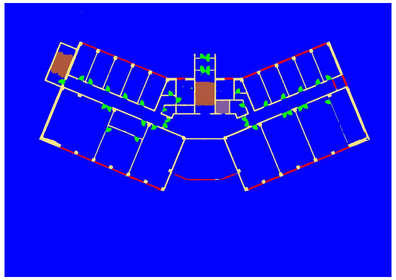
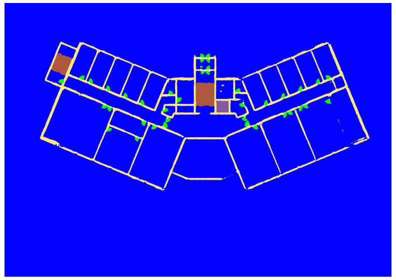
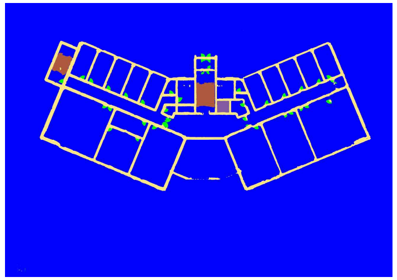
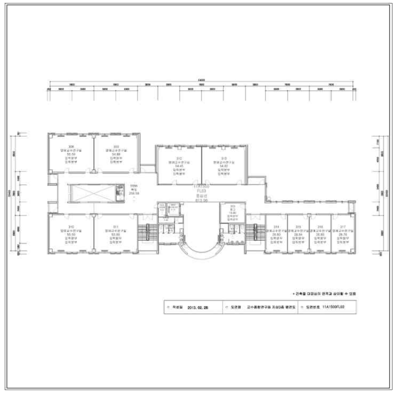
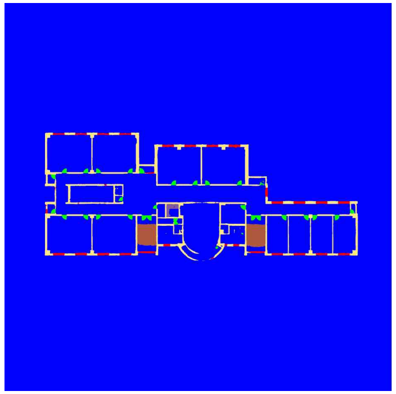
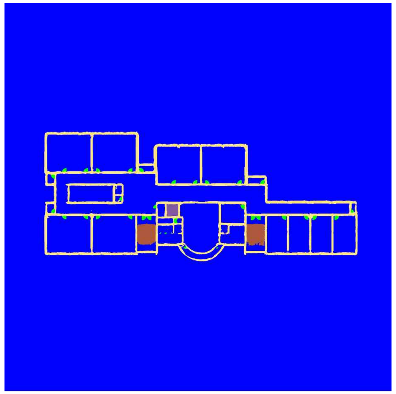
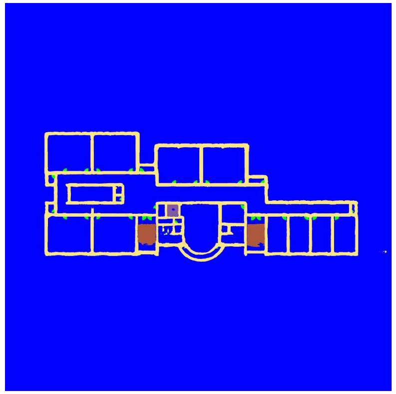
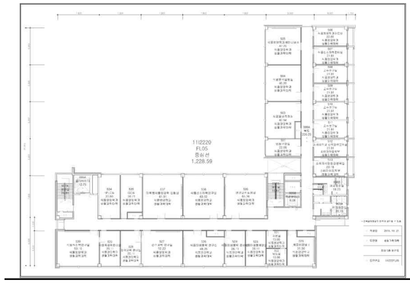
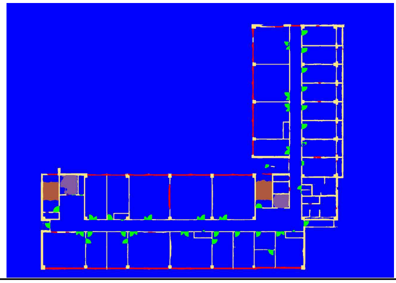
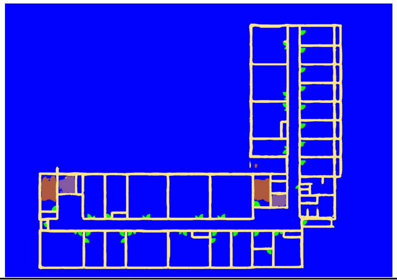
The purpose of the indoor structure map was to improve topology extraction performance, not to achieve high pixel-wise accuracy. Regarding the new annotation, it was important that: 1) the extracted wall expresses the clear division of the room, and 2) the changed spatial relationships between doors and walls are reflected for a better enclosure.

[Table 4-3] shows the results generated by each learning stage, for a total of seven test floor plans. The first column represents the input scanned floor plan and the second column shows the results of applying the pre-trained model described in section 3.1.1. In the proposed technique, transfer learning was performed in two stages: retraining the only classifier and fine-tuning the entire layer. The third column shows the result of the model, in which only the classifier was retrained while freezing the feature extractor. The last column shows the indoor structure maps generated by the final model, in which fine-tuning was performed for the entire layer after retraining the classifier.

[Table 4-3] Comparison between the generated indoor structure maps and the pre-trained model outputs

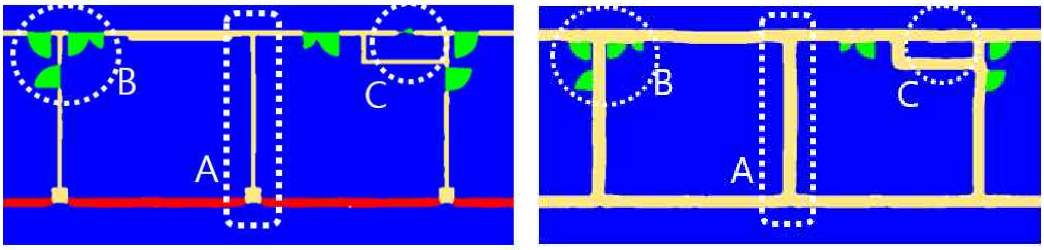
Input	Result of pre-trained model	Result of training classifier only	Result of fine-tuning
			
			

Input	Result of pre-trained model	Result of training classifier only	Result of fine-tuning
			
			

Input	Result of pre-trained model	Result of training classifier only	Result of fine-tuning
			
			
			

Comparing the results of the pre-trained model and the proposed model, the output of the proposed model was superior in that it divided the room through a neat, straight wall. In particular, protrusions from a wall, such as a column, can increase a room's structural complexity. The corresponding part of the wall was simplified in the proposed model (Part 'A' as shown in [Figure 4-1]). That is, the model preserved the overall structure of the room layout. Additionally, comparing the results of the model that retrained only classifier and the final model (with fine-tuning), the final model extracted this wall more successfully; it was much better enclosed. In addition, compared to the former model, the walls extracted by the final model were slightly thicker and were more simplified.

The relationships between walls and doors were modified in the new annotation so that the closure of each room could be more strongly learned. Specifically, each inner room was enclosed with only the walls, and each door was overlain onto the walls. The feature maps in the third and fourth columns show that doors were successfully extracted by reflecting this modification (Parts 'B', and 'C' in [Figure 4-1]). In terms of doors, there was no significant difference between results from the model with the retrained classifier and the final model.



(a) Result of pre-trained model (b) Result of the proposed model

[Figure 4-1] Comparison of extracted wall and door styles between the pre-trained and final model ('A' expresses the difference in protrusion of the wall, 'B' expresses the difference in relationships between door and wall, and 'C' expresses the difference in room division)

The fourth and fifth columns of [Table 4-2] summarize the results of visual interpretations considering the two criteria. The entire test set was extracted with integrated style walls, focusing on the structure of the space. The created indoor structure map also well reflected the deformation of the annotation to ensure the space enclosure. Several non-closed areas were found, as they were relatively small spaces or staircases with unclear boundaries. The enclosure of spaces with stairs is irrelevant regarding creation of the network model because additional staircase nodes can be generated using the extracted stair pixels.

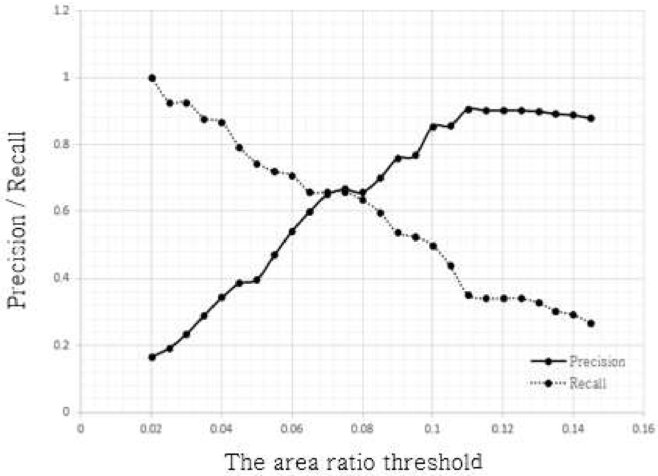
4.2.2 Results of topology retrieval

Indoor network models were created for seven test sets by applying the proposed procedures to the indoor structure maps presented in section 4.2.1. The network model can be created by end-to-end procedures, but the intermediate results of each step are also presented in this section to aid the interpretation of results. The result of set-G is described in this section, and the remaining results of all test floor plans are included in an appendix.

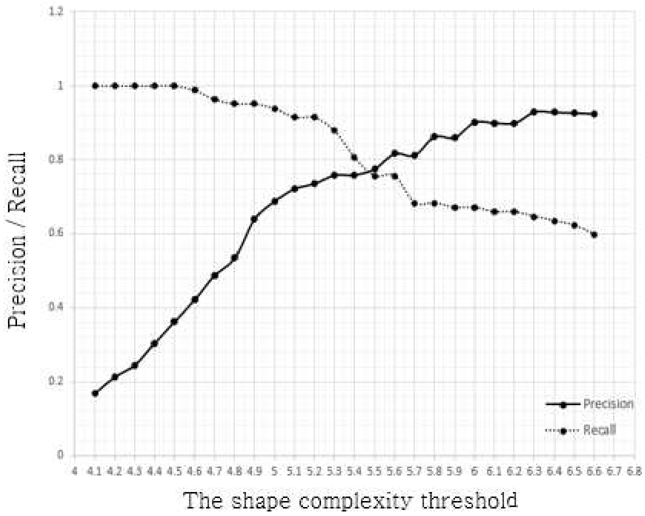
4.2.2.1 Classified horizontal transition spaces

First, corridors were detected as horizontal transition spaces. The generated indoor structure map included wall, door, elevator, and stair pixels. The feature map was separated for each feature to input them into the network model creation. Space polygons were created by detecting the area enclosed using the separated wall raster layer by wall pixels. The pixels of stairs and elevators were detected to generate bounding boxes.

According to the rules defined in [Table 3-4], enclosed spaces with high area ratio and shape complexity were detected as horizontal transition spaces. To find optimal thresholds for the rule, space classification was manually performed on floor plans in the new dataset to set reference data. 'True' was assigned to corridors considering the text labels of floor plans and their connections with other spaces. The area ratio and shape complexity were measured for all spaces, and precision and recall were calculated by matching the polygons detected as corridors with manually classified corridors. PR curves were plotted according to variations in given thresholds. The optimal thresholds were set by assigning a negative buffer based on the trade-off points of precision and recall. Out of 1,084 spaces, 82 were identified as corridors; PR curves for area ratio and shape complexity value are presented in [Figure 4-2]. The trade-off points for precision and recall, and for area ratio and shape complexity, were 0.06-0.07 and 5.4-5.5, respectively. Therefore, the reference values of area ratio and shape complexity for horizontal transition space detection were set as 0.06 and 5.4, respectively.



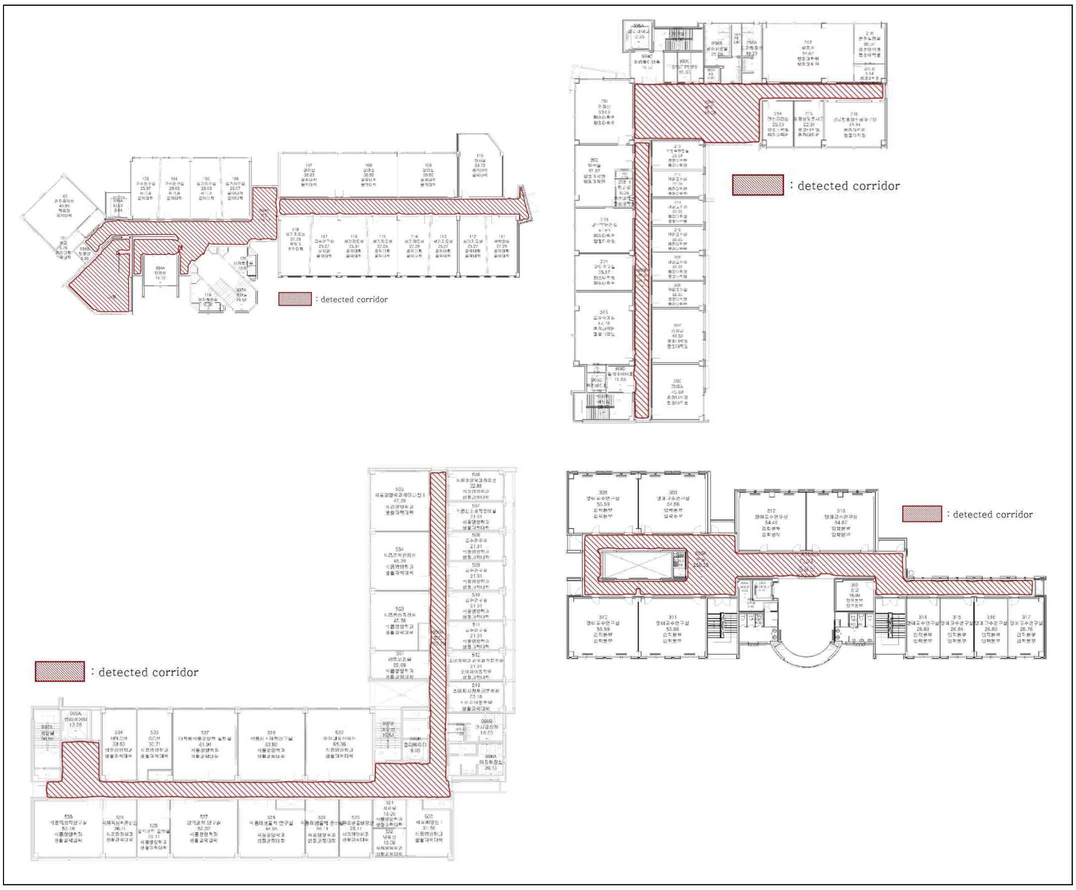
(a) The PR curve for area ratio



(b) The PR curve for shape complexity

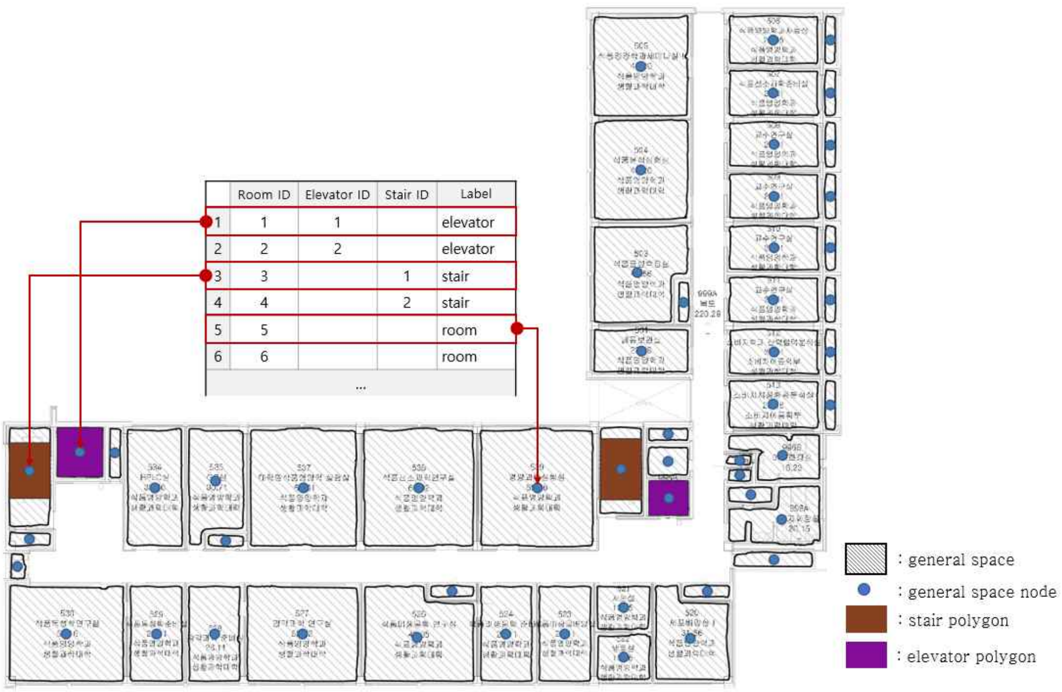
[Figure 4-2] PR curves for area ratio and shape complexity

[Figure 4-3] shows the results of detecting corridors for the test floor plans. Comparing the detected polygons with the text labels in the floor plans confirmed that corridors with various structures and shapes were detected appropriately. Therefore, 0.06 and 5.4 (thresholds for the area ratio and shape complexity, respectively) can be regarded as reasonable values. Two factors for corridor detection are invariant to the scale or size of the input floor plans since they are dimensionless values representing proportions. Also, as mentioned in 3.1.2 and 4.1, since the dataset consists of floor plans targeting buildings with various shapes in a wide range of sizes, it is suitable for extracting representative thresholds by setting the dataset as a sample. Therefore, the estimated thresholds can be used as reference values, even if the input dataset changes. Thresholds can be supplemented by adding a dataset with unique-shaped buildings or unusual cases of corridors to the sample to strengthen the representativeness of the thresholds.



[Figure 4-3] Detected corridors of test floor plans C, D, F, and G (in clockwise order from top left)

The nodes of general spaces with label attributes could be created concurrently while detecting corridors by using Procedure 1 ([Table 3-5]). [Figure 4-4] shows the resultant node layer of general spaces for set-G. The centroids of the rest spaces (except for corridor spaces) were extracted and saved as nodes with the 'room' label. Then, 'stair' and 'elevator' labels were stored in the label attributes to the nodes of rooms intersecting with stair and elevator bounding boxes, respectively.



[Figure 4-4] The created node layer of general spaces for set-G

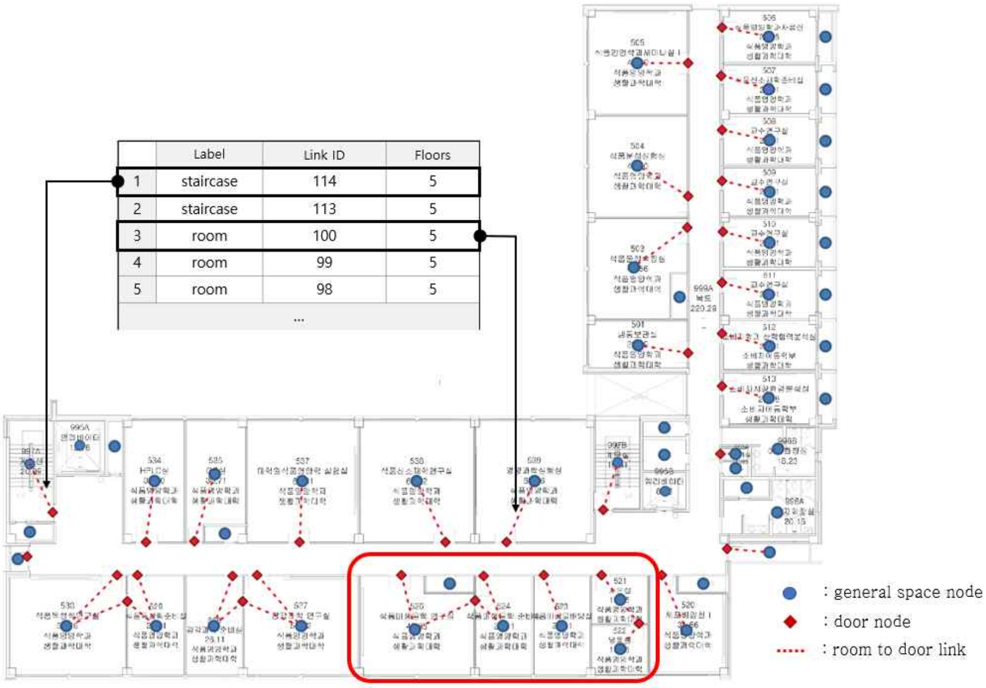
4.2.2.2 Results of links between general space and door

General spaces on each floor should be connected to a corridor through associated doors. Centroids were created as representative points by detecting doors in the extracted door raster layer. The door symbols are illustrated inside or outside of the room, depending on the opening direction. If the door symbol is drawn inside the room, it is easy to create a connection link because the corresponding door nodes are located inside the associated space. However, it was difficult to detect the related room when the door symbol was illustrated outside of the room, as said door was disjointed with the room. Therefore, the proposed Procedure 2 ([Table 3-6]) includes an adjustment of the door position to analyze the spatial relationship between doors and associated rooms. [Figure 4-5] shows the door nodes of set-G with and without this adjustment.

(a) Before adjusting door nodes (b) After adjusting door nodes

[Figure 4-5] The result of door adjustment for set-G

As a result of applying Procedure 2, door nodes that were snapped to the wall were created with the 'door' label, and 'room to door' links were created by connecting the door nodes to intersected room nodes ([Figure 4-6]). In other words, the nodes of each room were connected to their associated door node by the shortest distance. A few rooms were isolated because associated doors were not illustrated on the relevant floor plans. If stairs existed in a room with a door (i.e. a so-called staircase), the stair node was also connected to the associated door, as with room nodes. The links between stairs and doors were stored with the 'staircase' label, unlike other links stored with the 'room' ([Figure 4-6(a)]). Multiple links were created if there were multiple doors in a room. In addition, when a door existed between two rooms, a connection line between said two rooms and the corresponding door was created. This link between rooms were included in the network model for transitions without the need to go through a corridor. Inner rooms that were not directly in contact with a corridor were connected to outer rooms through a door ([Figure 4-6(b)]).



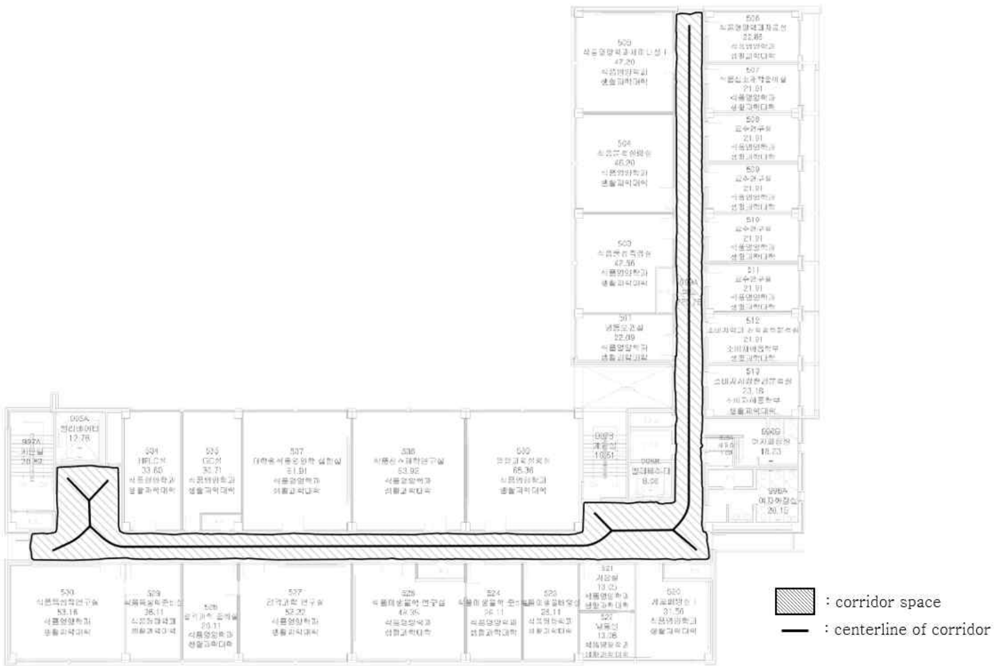
(a) The generated room to door links with ‘label’ attributes

(b) An enlarged view of room to door links

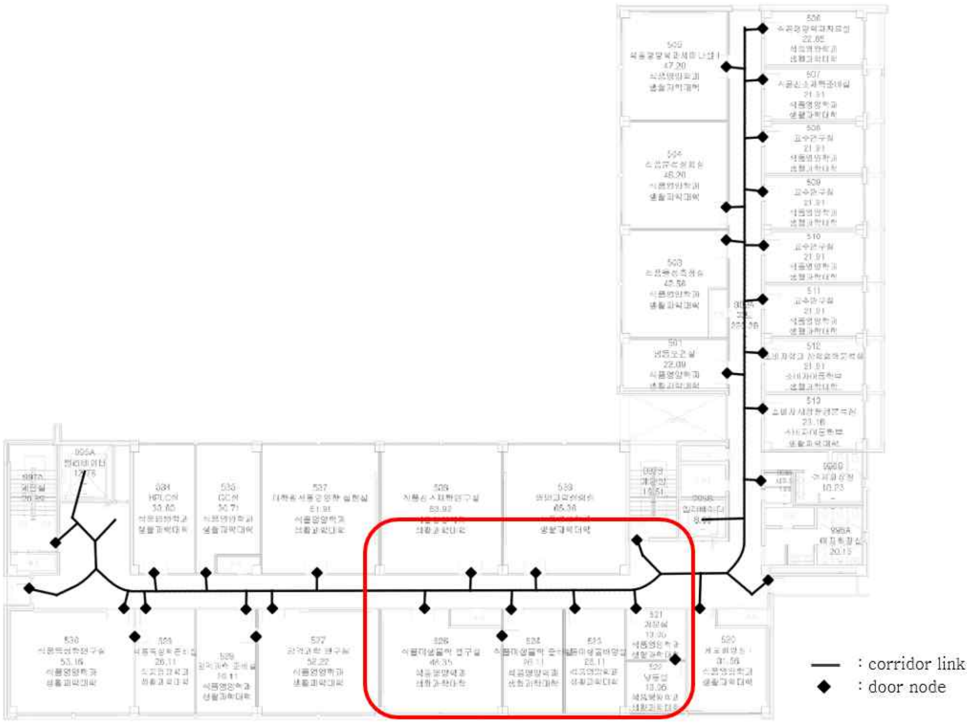
[Figure 4-6] The result of generating room to door links for set-G

4.2.2.3 Results of nodes and links for horizontal transition spaces

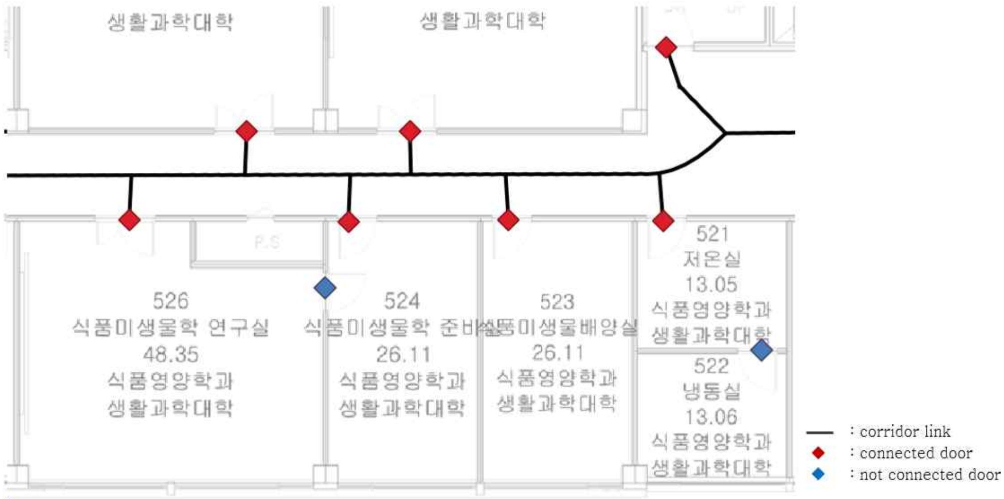
In the preceding process, each room was connected to the associated door based on the spatial relationship, and nodes and links were created for general spaces. Next, Procedure 3 ([Table 3-7]) was applied to generate nodes and links for corridor spaces. Centerlines were extracted as corridor links for corridor spaces, as shown in [Figure 4-7(a)]. Doors that were spatially joined with the corridor were selected and connected to the corridor links by the shortest distance ([Figure 4-7(b)]). The doors of inner rooms or doors located between rooms did not need to be connected to the corridor, and thereby connections were not added to the corridor links ([Figure 4-7(c)]). When every door was processed entirely, the intersections and endpoints of the corridor links were stored as corridor nodes ([Figure 4-7(d)]).



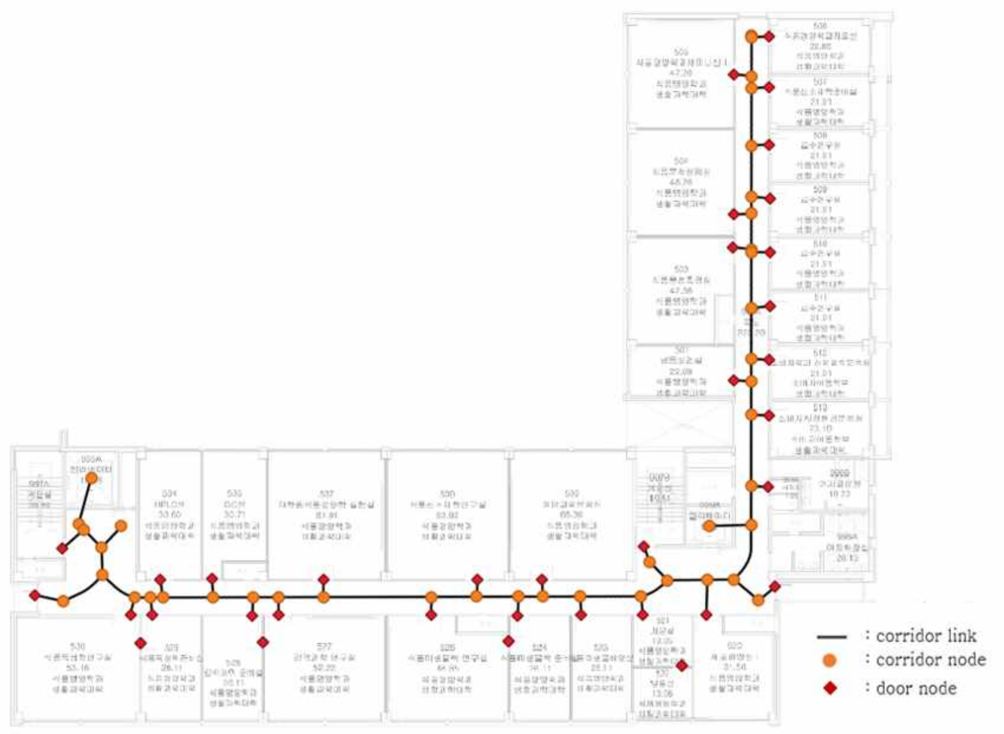
(a) Extracted centerline of the corridor space



(b) Result of connecting doors to the corridor link



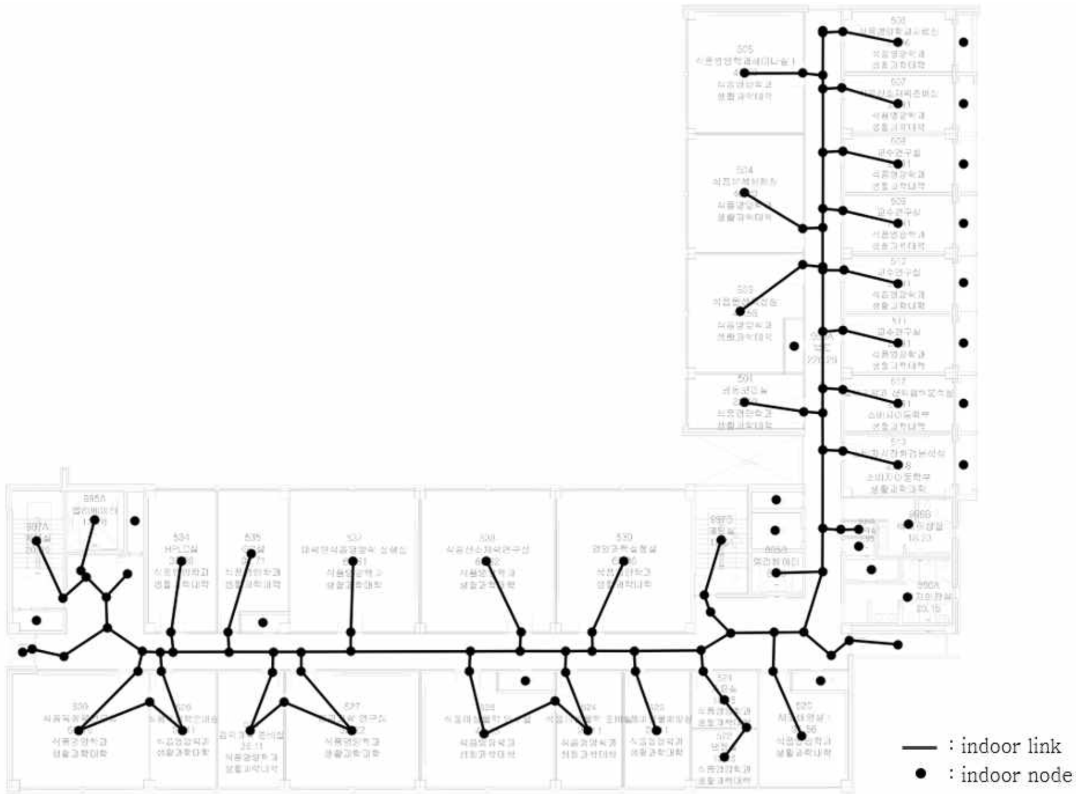
(c) An enlarged view of door to corridor connections



(d) Final result of corridor links and nodes

[Figure 4-7] The result of generating corridor links and nodes for set-G

Once corridor nodes and links were created entirely, those corridor nodes, previously created door nodes, and room nodes were then merged. Then, node ids were assigned to complete the final horizontal node layer. Likewise, the final horizontal link layer was constructed by combining and re-identifying all created links. Additionally, several bent segments were adjusted as straight links using two end nodes. The completed horizontal network model for set-G is represented in [Figure 4-8]. Each node and link were saved with the label attribute to allow for discrimination between types of indoor features.

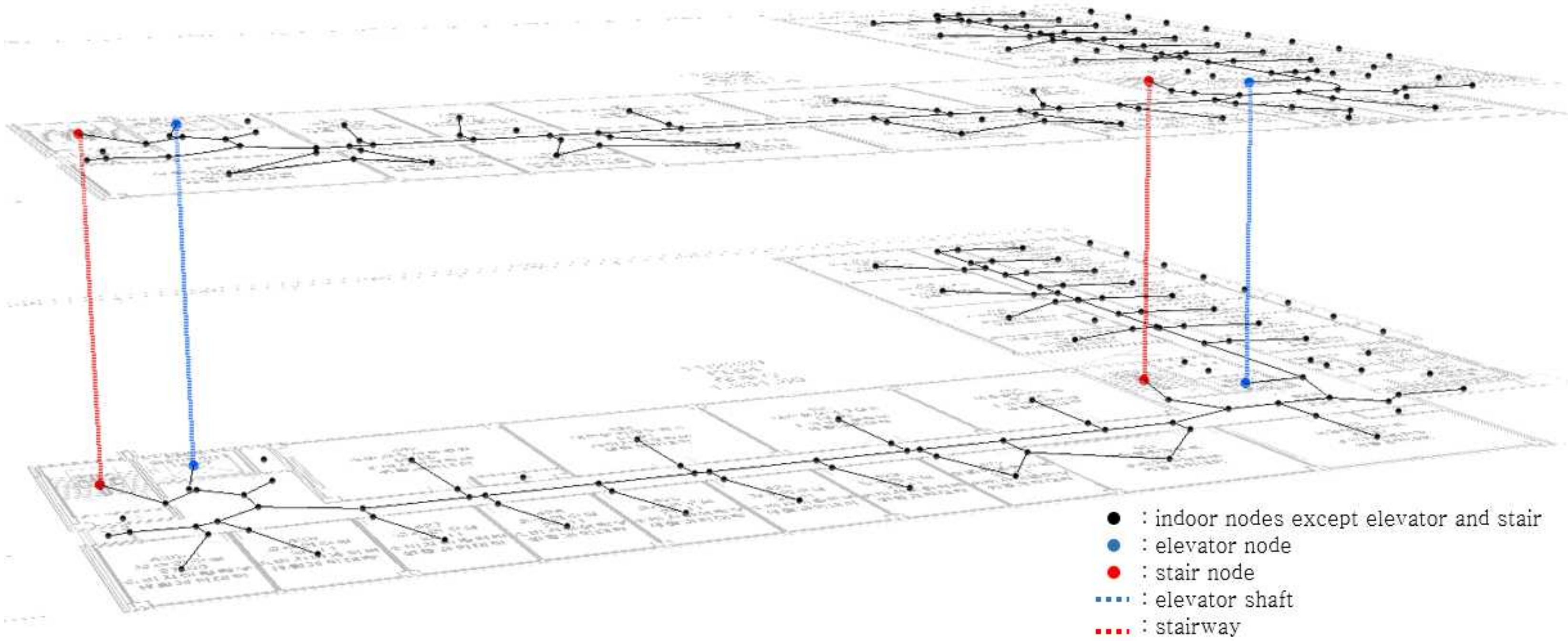


[Figure 4-8] Completed horizontal network model for set-G

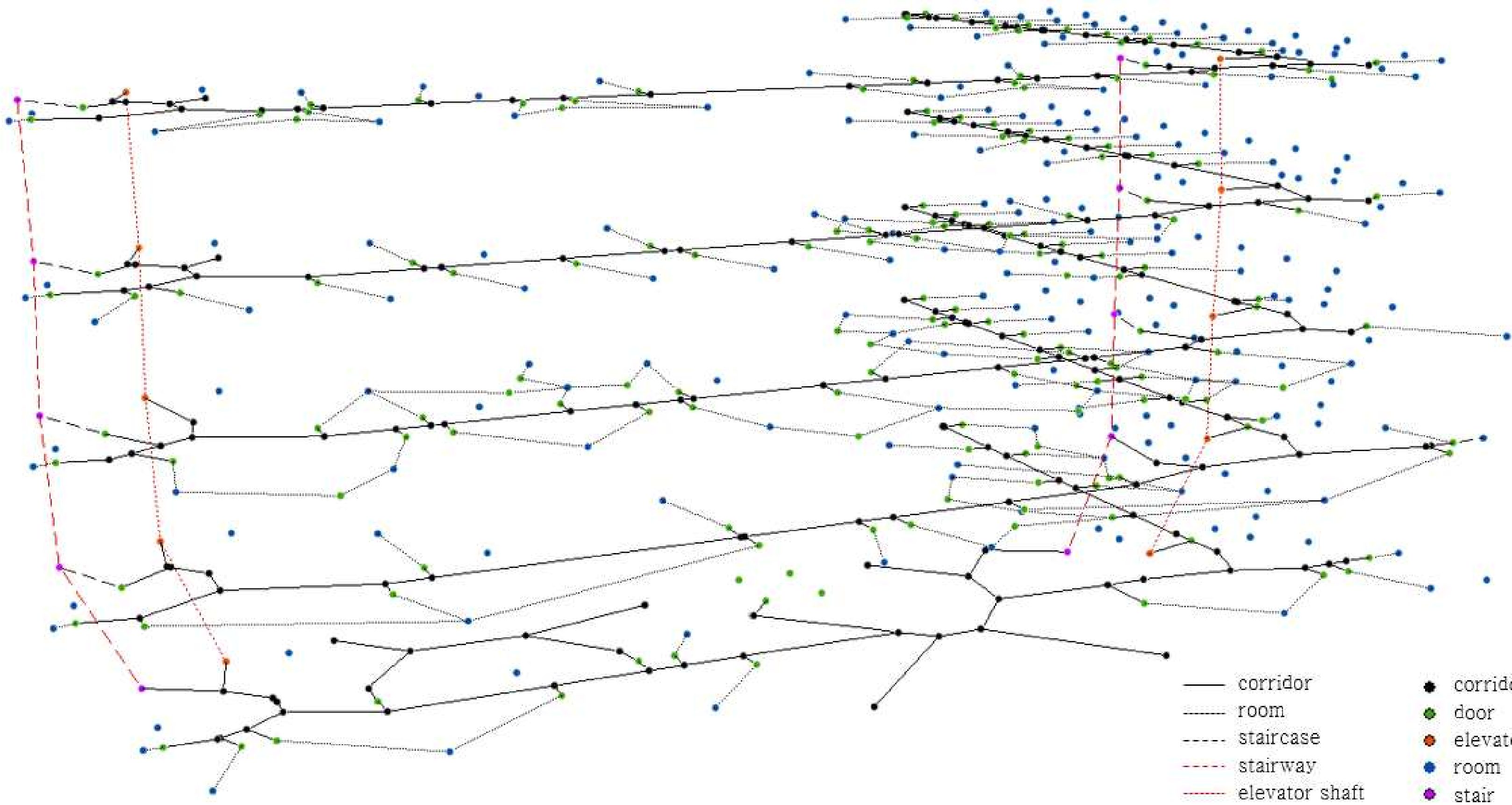
4.2.2.4 Results of vertical links

The multi-floor network model(presented as test building hereinafter) for set-G building, of which floor layout is neither simple nor unusual, was created to test the appropriateness of the vertical connecting procedure. It is adequate to test since there are multiple elevators and stairs in set-G. The horizontal network models for each floor were created through the above steps, then the inter-floor networks were created reflecting the vertical transitions, by applying Procedure 4 ([Table 3-8]). Matched pairs of elevators and stairs, which were located on successive floors, were determined based on positional and overlap similarities. Then, nodes of matched pairs were connected to establish vertical links ([Figure 4-9]). This procedure was applied in units of two consecutive floors from the first floor.

The result of creating all vertical links by iterating this procedure up to the fifth floor is presented in [Figure 4-10]. Generally, connected stairs and elevators in floor plans were illustrated in identical locations with the same scale. In this case, inter-floor links were created as vertically straight lines. However, as the first floor of the test building slightly differed in scale from the other floors, the vertical links were created as oblique lines. Although these links were not vertically straight, it was confirmed that the matched pairs to be connected were normally detected and connected. This shows that the proposed procedure works even if the scales of the scanned floor plans for each floor do not precisely match.



[Figure 4-9] The generated vertical links between fourth and fifth floor of set-G



[Figure 4-10] 3D visualization of the generated multi-floor network model of set-G

4.2.2.5 Connectivity generation result

As an essential characteristic for converting to a graph database, two nodes at both ends can represent a link segment. The result of storing two node-ids by detecting spatially intersecting nodes for each link is shown in [Figure 4-11]. The indoor network model included two link layers (horizontal and vertical), so connectivity information with nodes was generated for each layer.

	Label	Link ID	Start node ID	End node ID	Floors
1	room	92	5	53	5
2	corridor	31	51	71	5
3	corridor	18	52	72	5
4	room	91	52	118	5
5	staircase	114	56	90	5
...					

(a) The connectivity information of horizontal links

	Label	Link ID	Start node ID	End node ID
1	elevator shaft	1	97	91
2	elevator shaft	2	98	96
3	stairway	3	99	90
4	stairway	4	100	93

(b) The connectivity information of vertical links

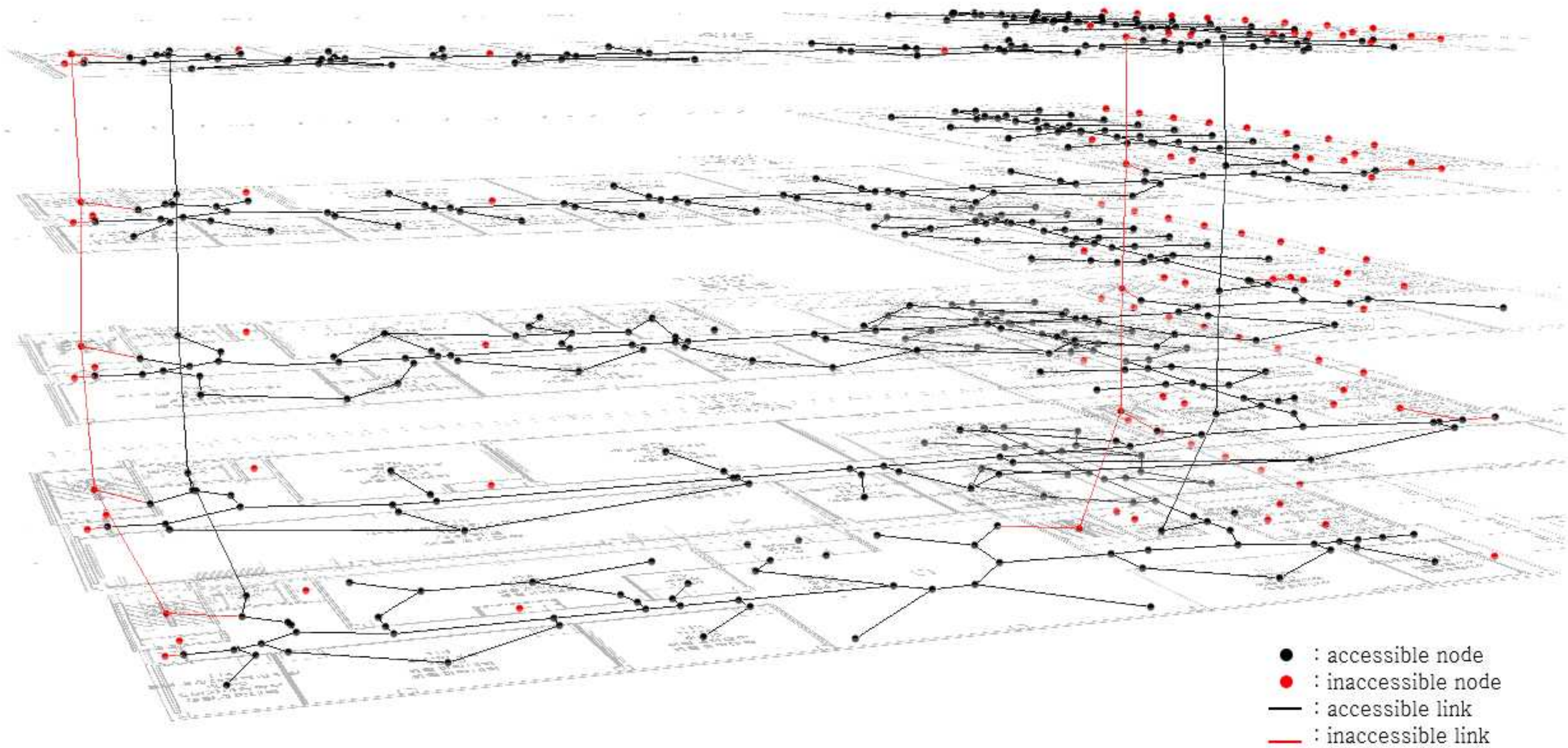
[Figure 4-11] Connectivity between nodes and links stored in the attribute table for set-G

4.2.2.6 Results of accessibility assessment

The accessibility assessment was conducted by applying the index suggested in section 2.3 to the created indoor network models. The factors of indoor features for the accessibility calculation were manually investigated by referring to the floor and elevation plans.

The calculated accessibilities were stored as attributes of nodes and links, and they can be used as quantitative cost. After calculating the quantitative accessibility values, each node was assessed as a Boolean value of inaccessible (T) and accessible (F) according to the thresholds presented in [Table 2-5]. These Boolean values were stored as the 'inaccessible' attribute.

Path selection criteria may vary depending on the passenger. For example, people, who consider movement convenience as the most critical factor, tend to choose routes with high accessibility. On the other hand, if efficient mobility is also considered, routes can be selected that reduce additional costs such as travel time, under the condition that routes are possible to pass through. The former case uses accessibility as a cost, whereas the latter case only requires the path to be planned using Boolean values. Therefore, both 'accessibility' and 'inaccessible' attributes need to be stored in the database. The results assessing nodes and links with regard to accessibility are visualized in [Figure 4-12].

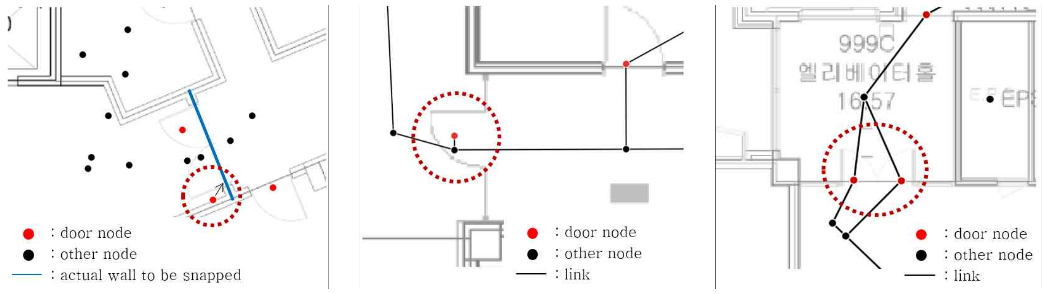


[Figure 4-12] 3D visualization of the multi-floor network model with accessibility for set-G

4.2.2.7 Qualitative evaluation

The creation error was analyzed through visual interpretation of the created indoor network models for the test sets. Specifically, causes of errors were examined for cases in which node or link creation was omitted or incorrectly created. Most of these errors were related to excluding nodes or links during construction; they were mainly caused by information that was not extracted during the step of generating indoor structure maps. In other words, when target objects were not included as extracted features in indoor structure maps, the nodes and links of the corresponding parts were inevitably omitted.

If the door location was adjusted improperly, errors occurred on room-to-door links ([Figure 4-13(a)]). It is difficult to recognize the wall that needs to be snapped to among two walls if the door exists at the point where said walls are orthogonal. Therefore, an error occurs in the process of adjusting the door location. Errors also occurred for floating door nodes that could not be connected to any room ([Figure 4-13(b)]). These floating doors were connected only to the corridor, and thus could not operate as openings. This occurred because the associated room in each case was not enclosed because of missing wall pixels, and so was merged into the corridor. Additionally, errors occurred whereby two individual nodes were created for a double-wing door ([Figure 4-13(c)]). This error occurred because the double-wing door was not dissolved as one door, since the annotation was created for a one-wing door symbol. However, it is expected that these errors could be mitigated by improving the trained model's performance to increase the accuracy of generating indoor structure maps.

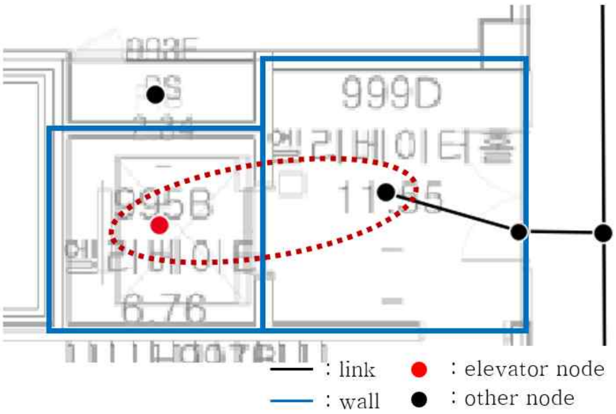


(a) Location error (b) Floating door node (c) Two nodes for double-wing door

[Figure 4-13] Error cases of generated door nodes

The most common error of room features was disconnected node creation. This occurred when the room-to-door links were not created because the associated door had an inconsistent symbol, and so could not be included in indoor structure maps. Similarly, nodes and links were also constructed for PIT rooms because PIT rooms, represented with X marks in floor plans, could not be distinguished from general spaces solely using walls. In short, untrained X marks caused this type of error. Corresponding cases could not be regarded as errors as the relevant doors and PIT rooms were not annotated in the new dataset. The network model's detail could likely be improved by training it with additional symbols (various types of door symbols, X marks, etc.) for the indoor structure map.

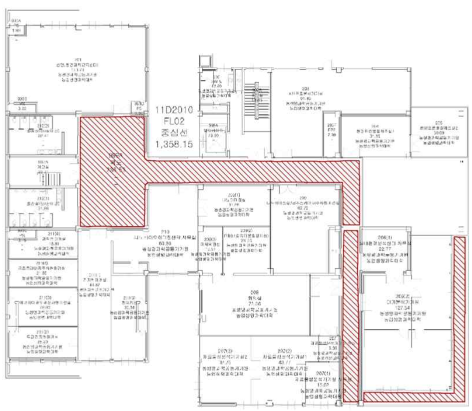
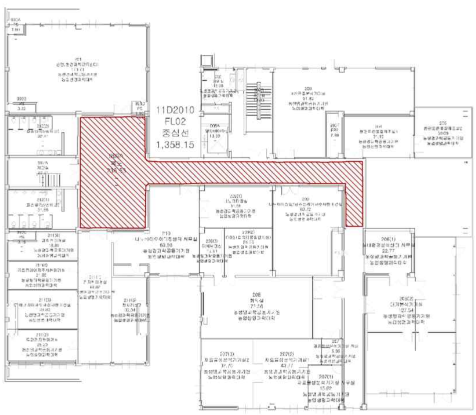
A few elevators were not connected to the corridor as shown in [Figure 4-14]. This occurred when the elevator hall existed as an individual space separated by walls. According to the rule of the proposed process, general spaces had to be connected to other spaces through a door. Therefore, elevator nodes had to be connected to an elevator hall through an associated door. However, this connection could not be generated as associated door nodes were not searched for. Similar to other features, it is expected that this issue could be solved by training elevators with more sophisticated annotation when creating the indoor structure map.



[Figure 4-14] Omission of links connecting the elevator and the corridor

As the sizes and shapes of buildings are diverse, and their floor layouts are also manifold, generalizing corridors' sizes and complexities is a limiting approach. In short, the presented thresholds could not completely include every type of corridor. For example, for general spaces with large areas exist, and where the corridor is relatively small, the corridor may have a smaller area ratio than the threshold ([Table 4-4]).

[Table 4-4] Detected corridors according to multiple thresholds

Case A (0.02, 5.4)	Case B (0.06, 5.4)
	

* Title of each cell represents (area ratio threshold, shape complexity threshold)

In a further study, various types of corridors should be used as reference data to improve thresholds. Based on the concept of EER, sub-networks of corridors can be created only for spaces that are reliably perceived as corridors by increasing thresholds. Conversely, the proposed rule is applied flexibly to create sub-networks for all possible spaces of corridors by lowering thresholds. In addition, more factors should be reflected in the rule to improve corridor detection accuracy. It is also necessary to determine whether to process large area rooms as corridors to create detailed network models. [Table 4-5] summarizes the types of error and their causes.

[Table 4-5] Error types of indoor network models

Feature	Error description	Reason of error
Door	Location error	If the door is located in the middle of ㄱ-shaped walls, an error may occur when snapping the door to the wall.
	Floating door node	If the associated room is merged into the corridor due to missing wall pixels, a floating door is created.
	Abstraction with two nodes for double-wing door	A double-wing door is not regarded as one door because of a failure in dissolving.
Room	Disconnected room node	The associated door is not extracted because it is illustrated with a different symbol from the annotation.
	Recognition of non-navigable rooms	Non-navigable spaces with X marks are unrecognized because space creation is completed only by walls.
Elevator	Omission of links connecting the elevator and the corridor	If the elevator hall is divided by walls, the connection link is not created because the associated door does not exist.
Corridor	Abstraction with only nodes as a general space for a few corridors	If the corridor is divided by doors, and the area ratio of each divided corridor is relatively small, the area ratio is smaller than the threshold value, so it may not be detected as a corridor.

4.2.2.8 Quantitative evaluation

In general, instead of accuracy evaluation, most previous studies related to network generation have conducted qualitative evaluation or performed distance comparison after the routing test using manually generated references and target data. Fu et al. (2020) evaluated their method by comparing route lengths from manual measurements with the achieved path network. Similarly, Teo & Cho (2016) performed the routing test and calculated relative error comparing to reference distance. Also, Hamieh et al. (2020) conducted scenario-based path computation to validate the utility of their resultant navigation model. Meanwhile, Yang & Worboys (2015) evaluated their result in different ways. They compared the created networks with sketch maps which ten participants draw to verify whether networks reflect significant functions and locations of indoor features in the human decision process. Clementini & Pagliaro (2020) only visualized the test network created using the proposed method, and Moritari et al. (2019) presented the time of graph construction with their suggested approach.

The network model represents 3D spaces and their relations by abstracting them into nodes and links. It is difficult to determine whether the network has been created accurately because the 3D spaces can be abstracted in various ways. In this regard, referring to the previous researches, scenario-based routing tests would be conducted to evaluate the target database generated through the proposed technique in the following section. Nevertheless, the quantitative evaluation was conducted on nodes that can be determined whether they have been appropriately created in this section. The evaluation results provide auxiliary information to determine the acceptable levels of models created for the purpose of conversion.

Kim (2014) evaluated the quality of generated pedestrian networks by comparing them with reference data using the four evaluation indices of Wiedemann (2003). These four indices are geometrical completeness, geometrical correctness, topological completeness, and topological correctness. As explained in the introduction, the core purpose of the network model created in this study was to retrieve topology, so accurate geometry generation was not included in the scope of the study. Therefore, the created indoor network models were evaluated in terms of the topological completeness and topological correctness (i.e., two of the four indices of Wiedemann (2003)).

Topological completeness indicates whether the nodes of the reference data, which have to be created, are included in the constructed target data. Topological correctness indicates the proportion of correctly generated nodes compared to the reference nodes. These indices are calculated as follows:

$$\text{Topological Completeness} = \frac{CB}{CR} \tag{4-1}$$

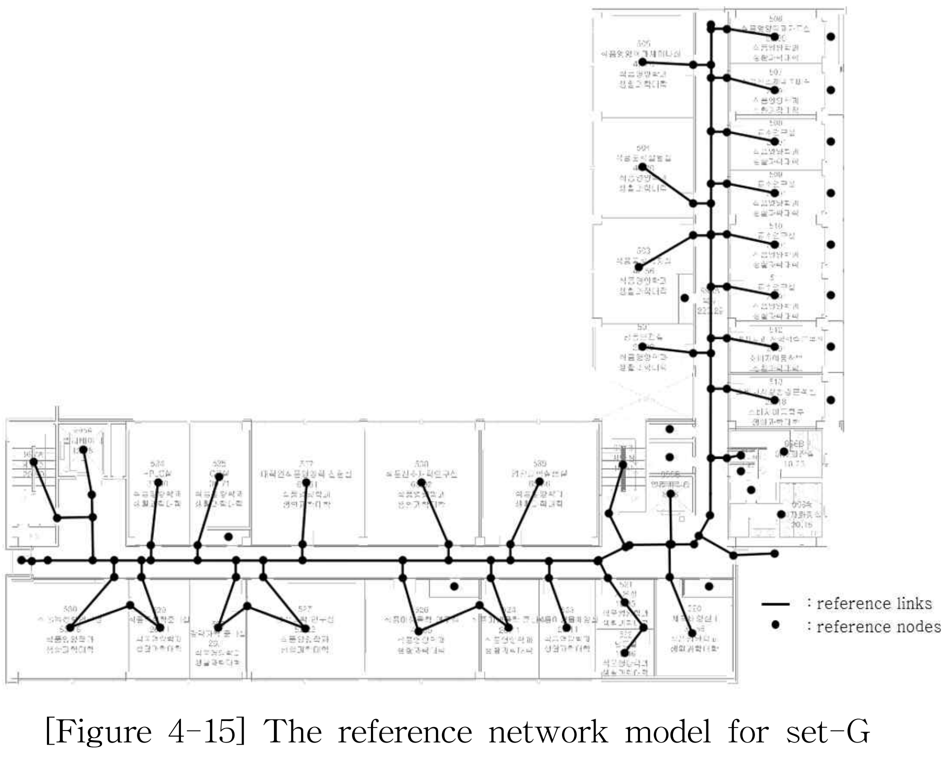
$$\text{Topological Correctness} = \frac{CB}{CG} \tag{4-2}$$

Where CB is the number of matched nodes between the reference and target data, CR is the total number of nodes in the reference data, and CG is the total number of nodes in the target data. Topological completeness and topological correctness are calculated as values zero or more and one or less. The annotated floor plan images were used as the reference data. The following target features were evaluated: room, stair, elevator, door, and corridor.

To evaluate the created network model with the presented indices, target nodes that match the reference nodes must be detected. Each object (room, stair, elevator, and door) in the reference data was polygonized, and target nodes intersecting these object polygons were determined as matched nodes. In the case of the corridor, matched nodes were determined after generating reference data according to the following criteria:

- Create nodes at both ends of the corridor centerline
- Create a node at the point where intersecting between doors and corridor centerline with the shortest distance
- Create centerlines by each subpartitioned area after dividing the corridor at the spot where the shape changes. Then, create nodes at that connecting point between adjacent centerlines

[Figure 4-15] depicts the reference network model for set-G. After manually inspecting the matched results, topological completeness and correctness were calculated, as shown in [Table 4-6].

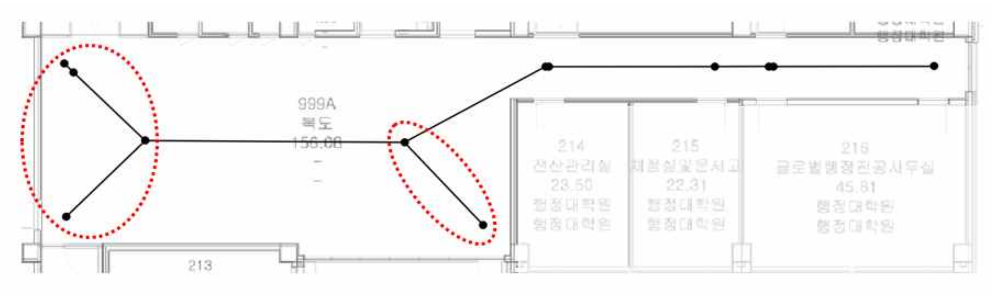


[Figure 4-15] The reference network model for set-G

[Table 4-6] Average topological completeness and correctness by feature of the test set

Feature	Node	
	Completeness	Correctness
Room	0.96	0.99
Corridor	0.95	0.77
Stair	1.00	1.00
Elevator	0.91	1.00
Door	0.97	0.98

Stairs represented by a unified and straightforward symbol in floor plans were generated as accurate nodes, without errors. On the other hand, some elevators were depicted with relatively small symbols on large floor layouts. Thus, the trained model often could not recognize them. Therefore, a few elevators were omitted, and the results indicated lower completeness compared to that of stairs. As corridors were abstracted using Voronoi diagrams in the proposed technique, corridors with similar widths and heights created links with wing edges. Consequently, more detailed nodes and links were generated than the reference corridor nodes ([Figure 4-16]). Therefore, the completeness was much higher than the correctness. The abstraction issue of the corridor with wing edges and the issue of undissolved double-wing doors ([Figure 4-13(c)]) are related to the details of the network. Although both issues lower the simplicity of feature abstraction, modification is not necessarily required since they can describe indoor features in detail.



[Figure 4-16] Example of a detailed network in a corridor

[Table 4-7] shows the completeness and correctness values for each test set. The average accuracy (*F-measure*) of test network models was 93% in this study. Sets-D -G, consisting of only regular straight walls and a general-shaped corridor, respectively, showed relatively high completeness and correctness. Although set-F has an atypical floor layout, both completeness and correctness were relatively high since almost every room was extracted as enclosed spaces. Set-A, which included the simplest and most standardized spaces, had a completeness of 1, and its nodes were well generated without errors for all spaces. However, corridor links connecting stairs to corridors were created because the stairs were in corridors, rather than being divided into rooms (staircase). Therefore, more corridor nodes (for connections with stairs) were extracted, which thus resulted in relatively low correctness. Overall, completeness was higher than correctness because more nodes were extracted than reference data, due to the influence of the corridors as described above.

Indoor nodes created using crowdsourced data in Kim *et al.* (2015) were evaluated using precision and recall, and they represented an average accuracy of 90%. Although there is difference in target environment with this study, pedestrian networks created by Kim (2014)'s technique were verified using an identical evaluation index, and the average accuracy was 85%. Referring to the accuracy of networks in previous researches, network models created using the proposed technique with the accuracy of 93% were acceptable to convert to the target database.

[Table 4-7] Average topological completeness and correctness by the test set

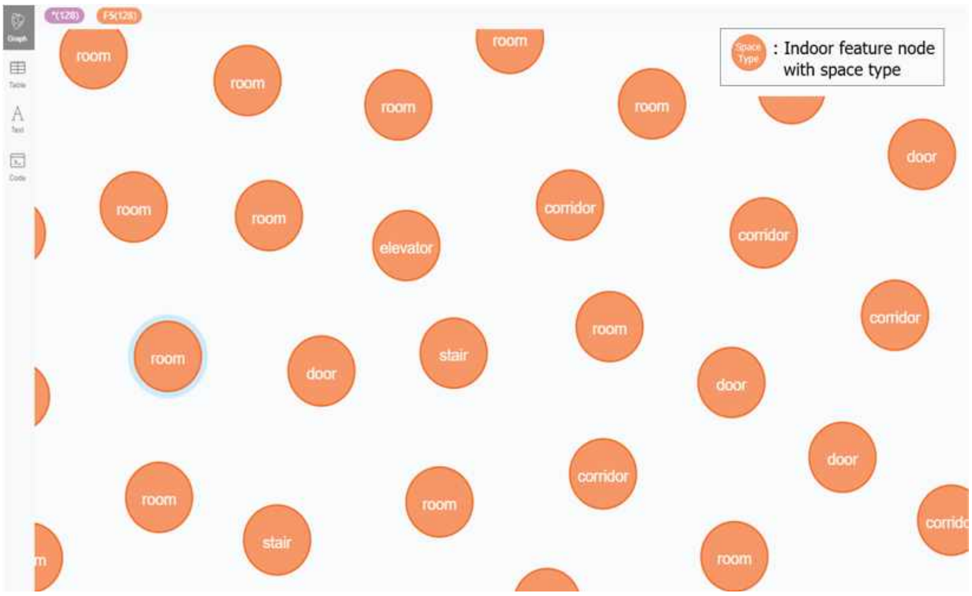
Building-Floor	Node	
	Completeness	Correctness
A	1.00	0.90
B	0.95	0.89
C	0.95	0.87
D	1.00	0.95
E	0.96	0.83
F	0.93	0.94
G	0.97	0.96

The quality of connectivity information is more critical because the key purpose of the network model is its ability to convert into the graph database. The ratio of disconnected nodes, excluding PIT rooms, of the network model was on average 9% per test set, following the connectivity validation of Khalili & Chua (2015). Thus, it appears that the created network models were horizontally connected. It was also confirmed that all floors were vertically connected. That is, network models represents horizontally and vertically connected graphs between spaces in order to be converted into a graph database. Rather than a quantitative evaluation, this study focuses on verifying the utility of the created network model for various applications such as routing. The database's usability regarding construction using the proposed technique is verified further in the next section.

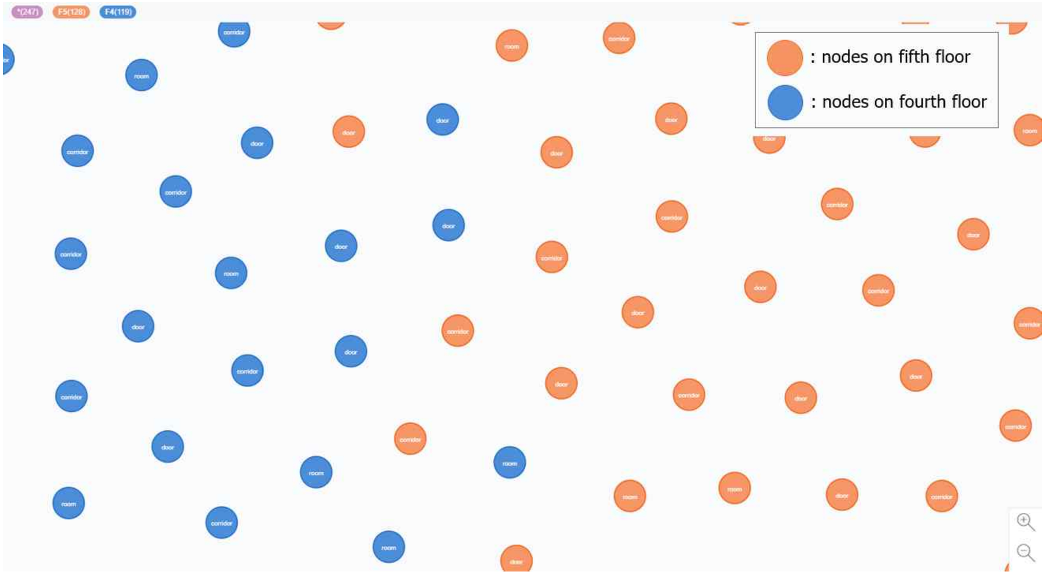
4.3 Generated indoor graph database for PWMD

4.3.1 Results of the indoor graph database for PWMD

The created indoor network models were converted into an indoor graph database through Procedure 5 ([Table 3-10]). First, the indoor node layers in network models were imported as groups of graph nodes ([Figure 4-17(a)]). The label 'floors' was designated to query each floor ([Figure 4-17(b)]). 128 nodes of set-G, the exact same number as that of the network model, were imported into Neo4j.



(a) Imported indoor node layer in Neo4j



(b) Imported indoor node layer with label in Neo4j

[Figure 4-17] The result of importing indoor node layer in Neo4j

A set of Cypher commands was created to form relationships using the link layers in the network models and the set properties between the imported nodes. The connections among nodes were generated as bidirectional relationships, as people can move in both directions within indoor environments. The 'inaccessible', 'accessibility', and 'distance' attributes were allocated as relationship properties to optimize route extraction for PWMD. Commands for indoor graph database conversion were then automatically written using only the link layer in the network model. [Table 4-8] presents examples of the Cypher commands used to configure relationships to support horizontal and vertical transitions in the test building. These commands were written to create a relationship between nodes after searching the imported nodes having the same id as both nodes in each link segment.

[Table 4-8] The example of Cypher commands for configuring relationships

```
# Cypher commands for relationships of horizontal connection
MATCH (n1:F5) WHERE n1.N_ID=20
MATCH (n2:F5) WHERE n2.N_ID=44
CREATE (n1)-[r1:corridor {accessibility: 157.25, inaccessible: 0, distance: 55.80}]->(n2)
CREATE (n2)-[r2:corridor {accessibility: 157.25, inaccessible: 0, distance: 55.80}]->(n1);
MATCH (n1:F5) WHERE n1.N_ID=5
MATCH (n2:F5) WHERE n2.N_ID=62
CREATE (n1)-[r1:room {accessibility: 313.38, inaccessible: 0, distance: 78.75}]->(n2)
CREATE (n2)-[r2:room {accessibility: 313.38, inaccessible: 0, distance: 78.75}]->(n1);
MATCH (n1:F5) WHERE n1.N_ID=57
MATCH (n2:F5) WHERE n2.N_ID=93
CREATE (n1)-[r1:staircase {accessibility: 315.53, inaccessible: 1, distance: 102.51}]->(n2)
CREATE (n2)-[r2:staircase {accessibility: 315.53, inaccessible: 1, distance: 102.51}]->(n1);

# Cypher commands for relationships of vertical connection
MATCH (n1:F4) WHERE n1.N_ID=98
MATCH (n2:F5) WHERE n2.N_ID=96
CREATE (n1)-[r1:elevatorShaft {accessibility: 30.0, inaccessible: 0, distance: 30}]->(n2)
CREATE (n2)-[r2:elevatorShaft {accessibility: 30.0, inaccessible: 0, distance: 30}]->(n1);
MATCH (n1:F4) WHERE n1.N_ID=99
MATCH (n2:F5) WHERE n2.N_ID=90
CREATE (n1)-[r1:stairway {accessibility: 30.0, inaccessible: 1, distance: 30}]->(n2)
CREATE (n2)-[r2:stairway {accessibility: 30.0, inaccessible: 1, distance: 30}]->(n1);
```

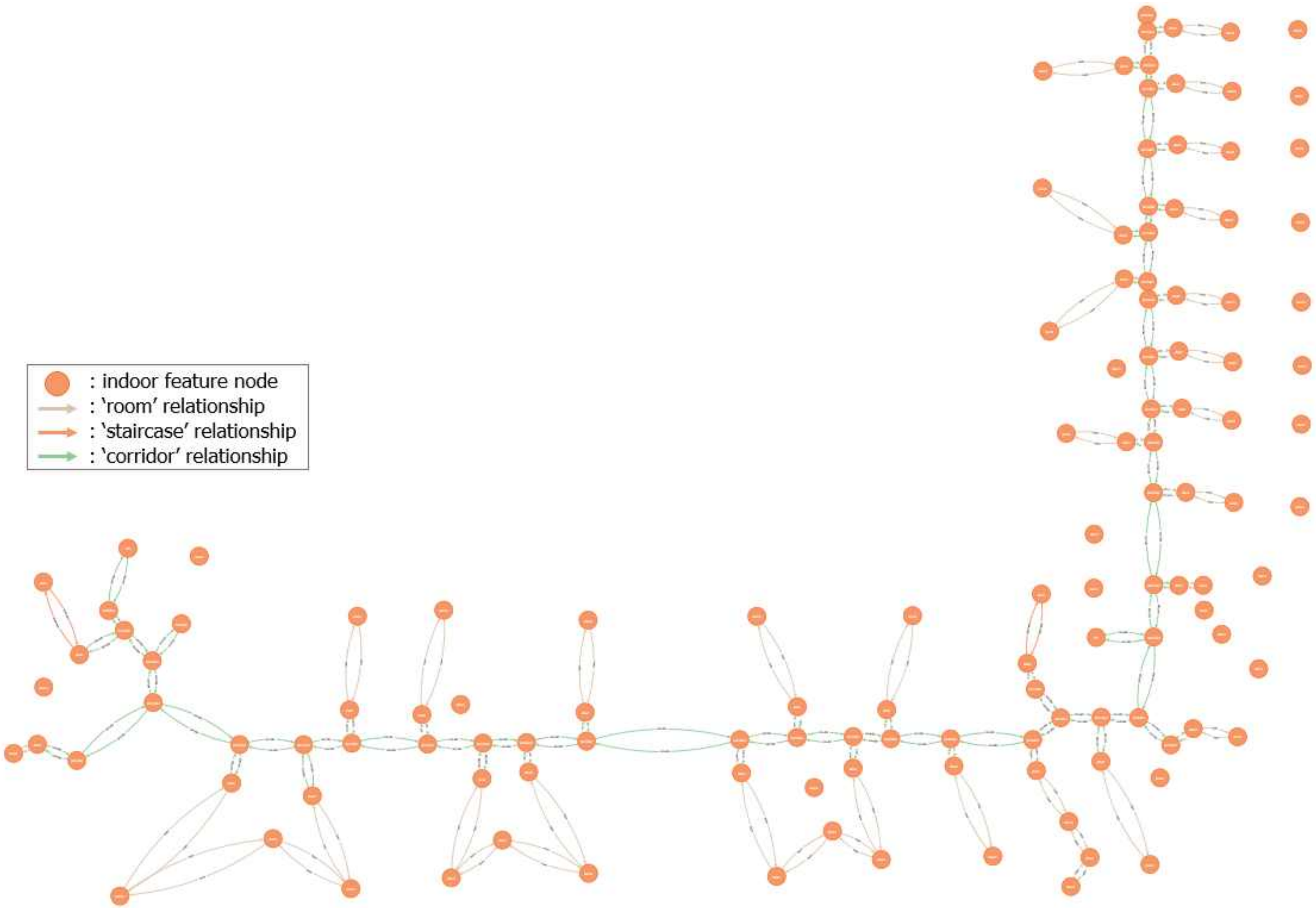
The conversion to the indoor graph database was completed by executing the written Cypher commands on the Neo4j desktop. [Table 4-9] shows the converted indoor graph database for the test sets and [Table 4-10] shows the conversion results for the test building, including set-G. The indoor graph databases for the test sets contained nodes and links for horizontal movement, such as rooms, doors, and corridors. Named relationships for stairways and elevator shafts were also created for vertical movement. [Figure 4-18] presents the indoor graph database for set-G and [Figure 4-19] shows those with vertical connections for the test building. Through a simple 'MATCH' query, the inner-floor and inter-floor graph can be searched, and the sub-graph of the desired area can be displayed as well ([Figure 4-19 (c)]).

[Table 4-9] The result of indoor graph database for test set

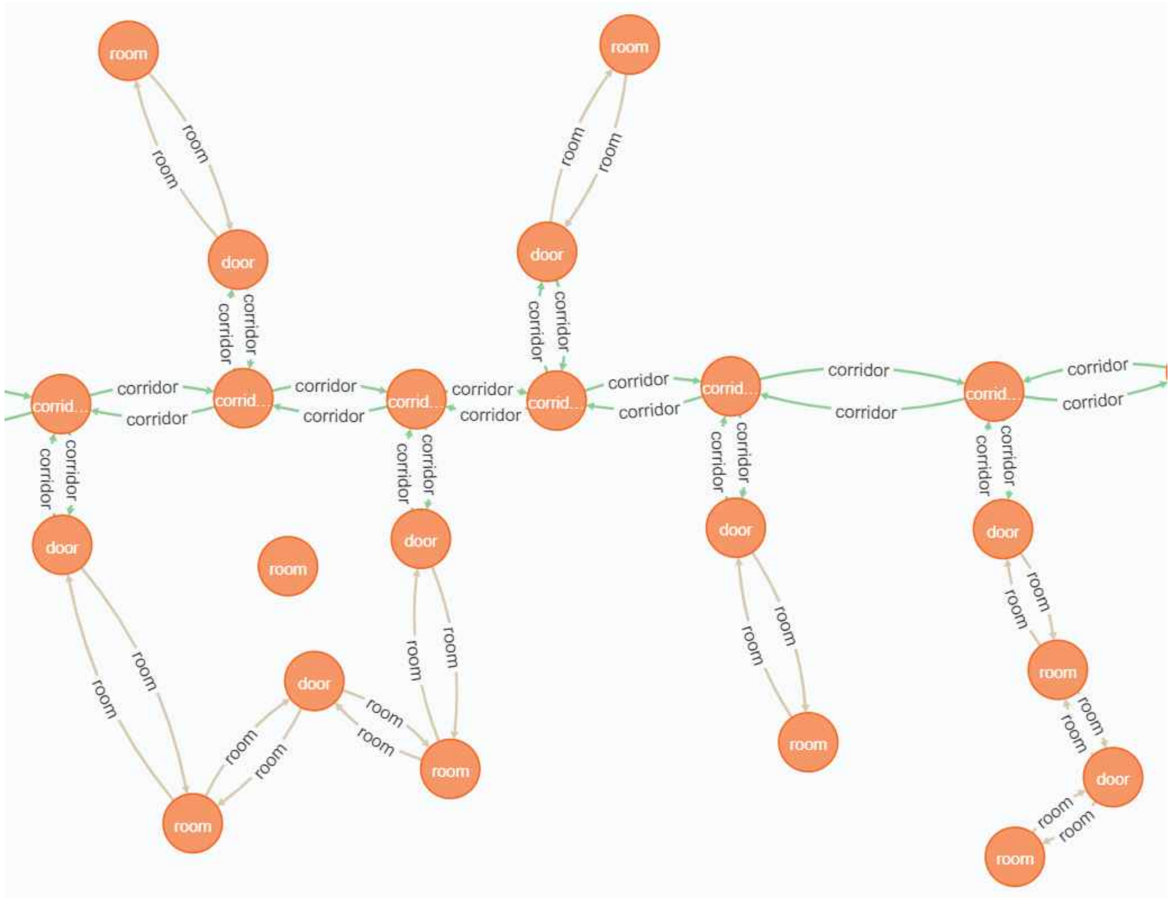
Test set	The number of nodes		The number of relationships
	Horizontal connection	Vertical connection	
A	68	1	134
B	182	5	380
C	91	2	176
D	89	4	170
E	104	3	212
F	92	3	188
G	124	4	220

[Table 4-10] The result of indoor graph database for test building

Floors	The number of nodes		The number of relationships	
	Horizontal connection	Vertical connection	Horizontal connection	Vertical connection
1	96	4	172	8
2	99	4	186	8
3	125	4	244	8
4	111	4	194	8
5	124	4	220	8

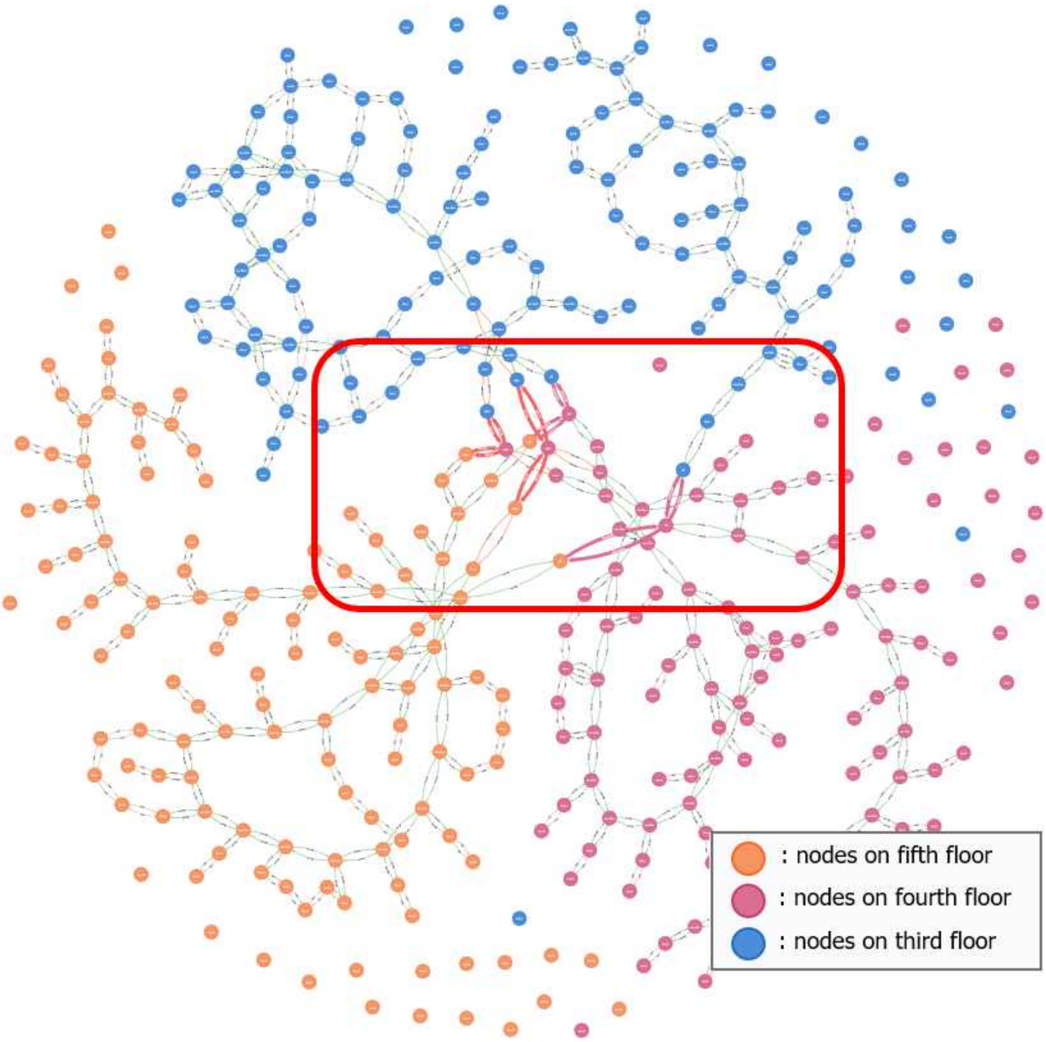


(a) The converted indoor graph database for set-G (fifth floor of test building)

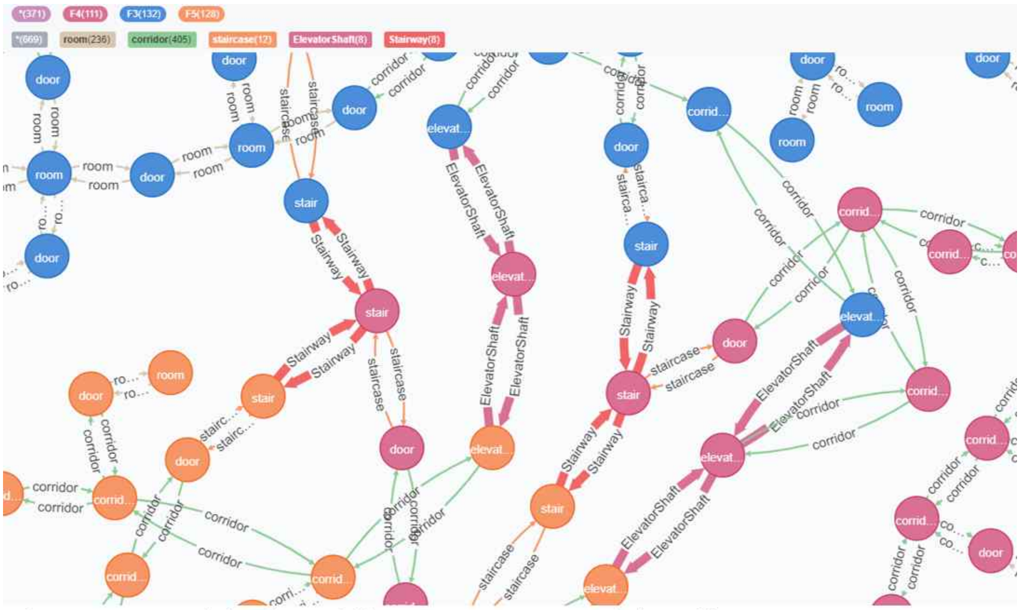


(b) An enlarged view of the indoor graph database for set-G

[Figure 4-18] The result of the indoor graph database for PWMD

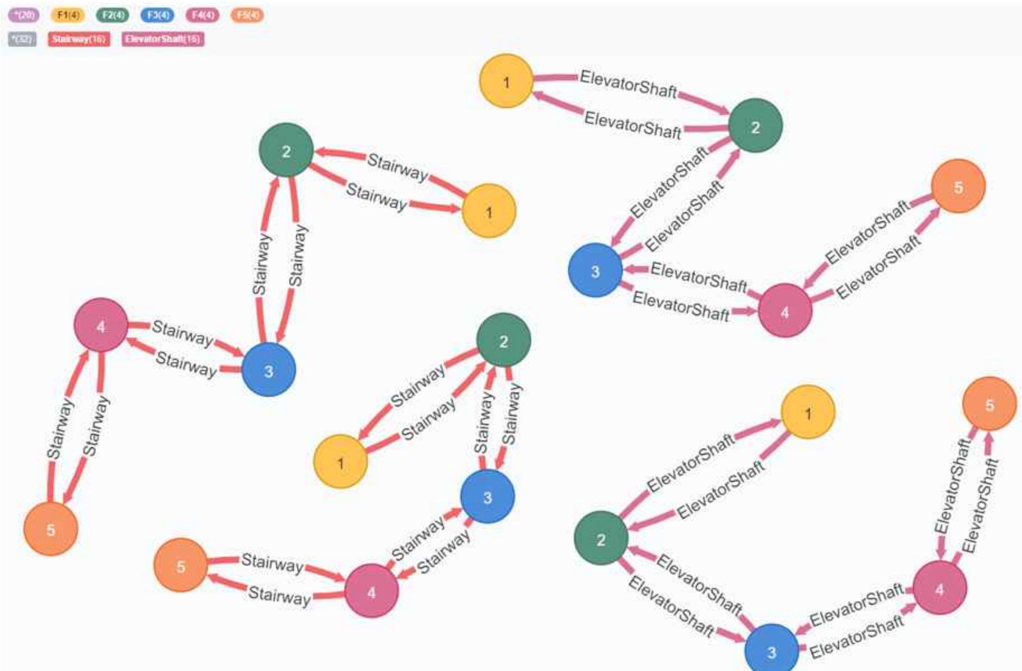


(a) With conversion of vertical links



* Arrows represent relationships, and thicker arrows represent vertical transitions

(b) An enlarged view of vertical links conversion



* The text in nodes represents the number of floors

(c) Sub-graphs of vertical connections from the first to fifth floors

[Figure 4-19] Converted indoor graph database with vertical connections for the test building

4.3.2 Query-based routing

Two scenario-based routings were performed to validate the proposed schema and the generated indoor graph database. These two scenarios were: (1) multi-floor routing, and (2) integrated indoor-outdoor routing. Arbitrary nodes in the graph were selected as the start and destination for routing, and path calculation was performed by applying the ONALIN[8] (Dudas *et al.*, 2009; Karimi & Ghafourian, 2010). The ONALIN was implemented in the following two steps; [Table 4-11] shows an example of a routing query.

1) Extracting feasible indoor graphs for PWMD by projecting only graphs with the property of 'false inaccessible'

2) Calculating shortest paths for feasible indoor graphs using the Dijkstra algorithm

Routing was performed using the Graph Data Science (GDS) library 1.1.3 for Neo4j with an identical experiment setup to that described in 4.1. All routing queries were executed within 10ms.

---

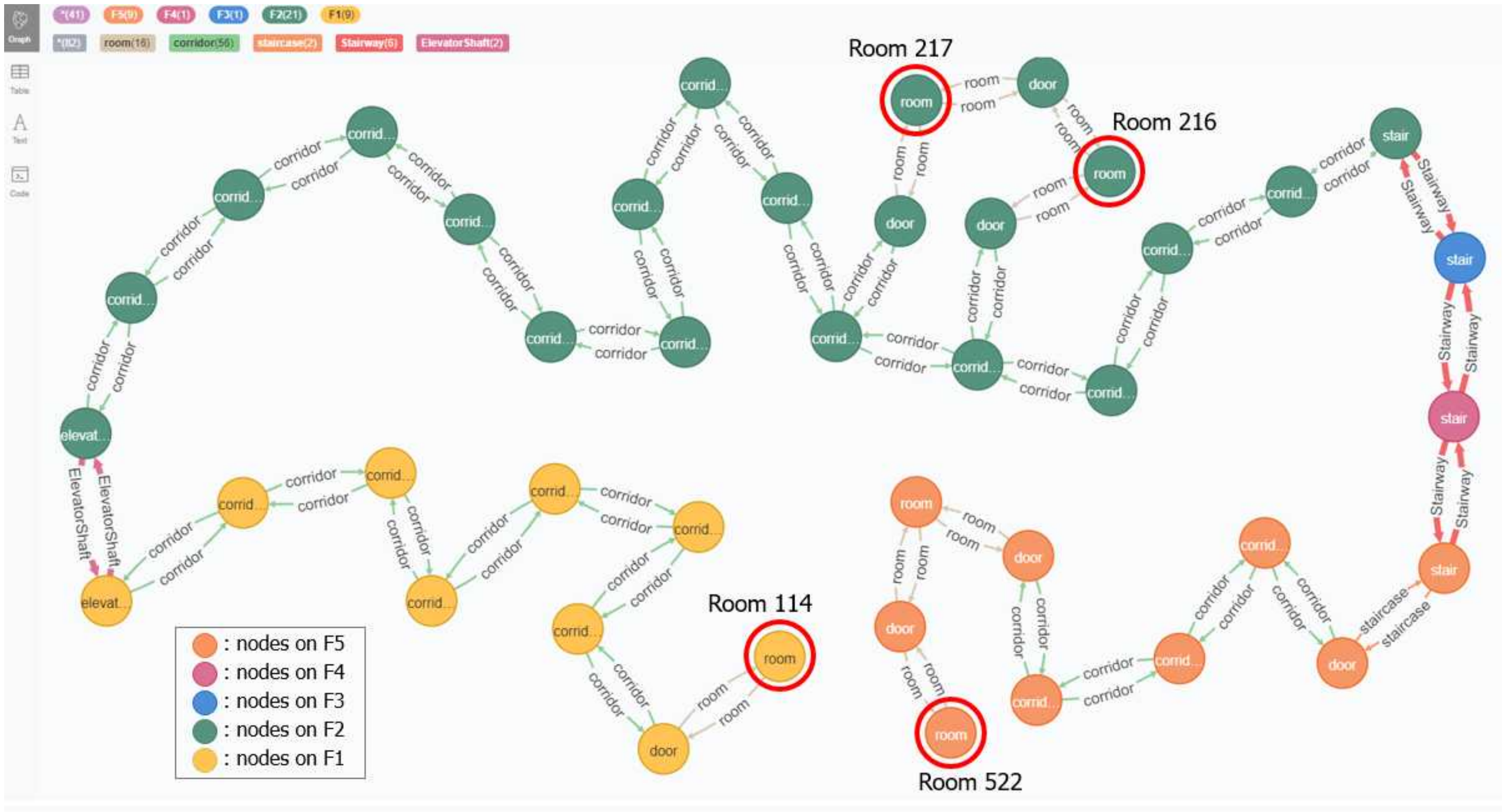
8) ONALIN represents ontology and algorithm for indoor routing; it consist of ONALIN-FN for searching feasible network and ONALIN-PR for deriving preferred routes.

[Table 4-11] The example of multi-floor routing query in Neo4j

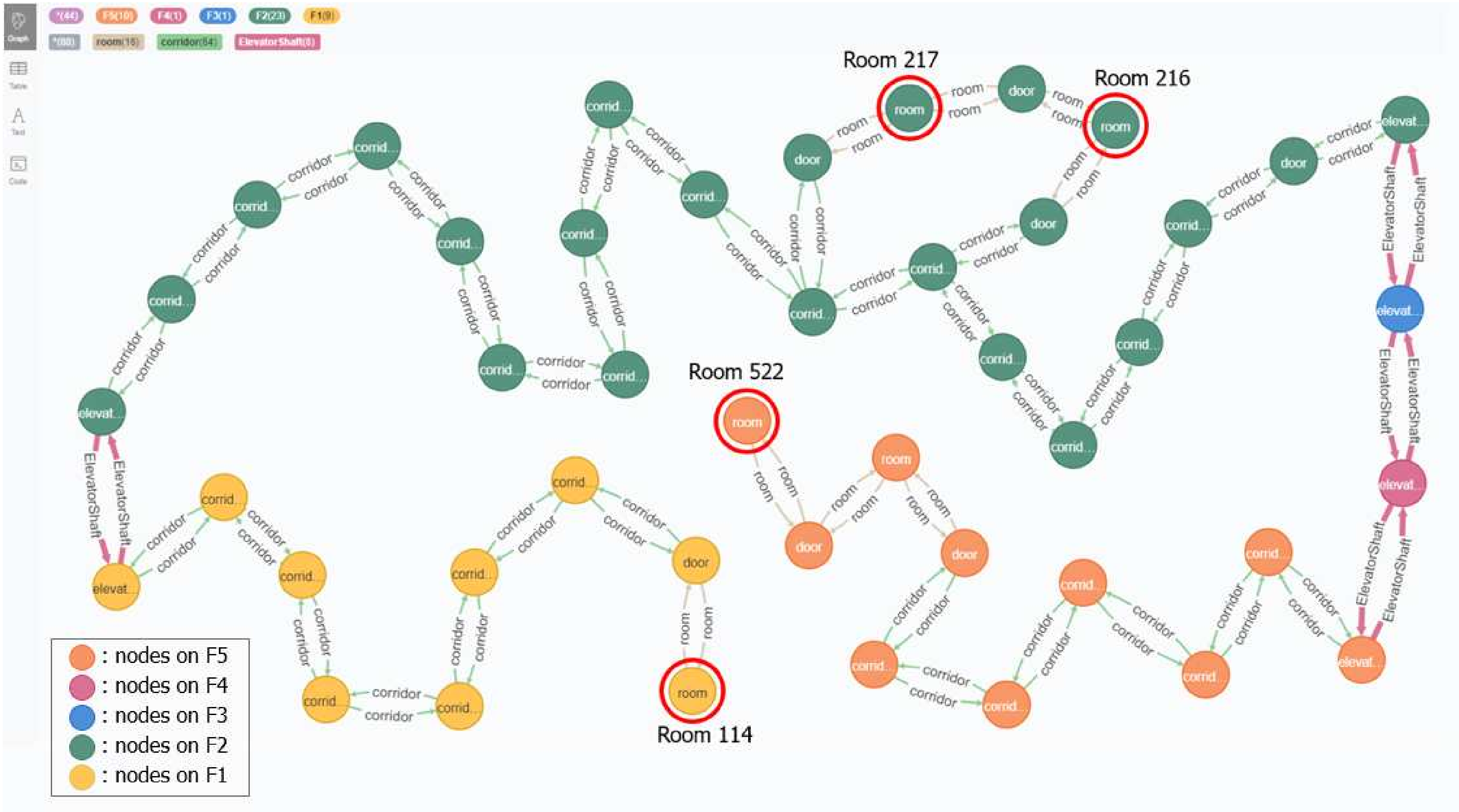
```
MATCH (start:F5 {N_ID: 9})
MATCH (end:F2 {N_ID: 5})
CALL gds.alpha.shortestPath.stream({
  nodeQuery:'MATCH(n) RETURN id(n) AS id',
  relationshipQuery:'MATCH(n)-[r:Corridor|Room|Staircase|Stairway|
ElevatorShaft {inaccessible:0}]->(m) RETURN id(n) AS source, id(m) AS
 target, r.distance AS weight',
  startNode: start,
  endNode: end,
  relationshipWeightProperty: 'weight'
})
YIELD nodeId, cost
WITH gds.util.asNode(nodeId) as nodes, cost
RETURN nodes.N_ID as NODE_ID, cost, nodes
UNION ALL
MATCH (start:F2 {N_ID: 5})
MATCH (end:F1 {N_ID: 5})
CALL gds.alpha.shortestPath.stream({
  nodeQuery:'MATCH(n) RETURN id(n) AS id',
  relationshipQuery:'MATCH(n)-[r:Corridor|Room|Staircase|Stairway|
ElevatorShaft {inaccessible:0}]->(m) RETURN id(n) AS source, id(m) AS
 target, r.distance AS weight',
  startNode: start,
  endNode: end,
  relationshipWeightProperty: 'weight'
})
YIELD nodeId, cost
WITH gds.util.asNode(nodeId) as nodes, cost
RETURN nodes.N_ID as NODE_ID, cost, nodes
```

Case 1. Multi-floor routing

Routing, including inter-floor movements, was assumed using the data from the first to the fifth floors of the test building. Starting from the laboratory (Room 522), which was located on the fifth floor, the optimal route to the first floor laboratory (Room 114) was derived via the preparation room (Room 217) and the studio (Room 216) on the second floor. Laboratory 522 is a room inside another room, and Rooms 217 and 216 are connected to each other with a door between these rooms. Referring to Park *et al.* (2020), it was assumed that the travel time through stairs and elevators was identical for intuitive confirmation of obstacle avoidance in each path. [Figure 4-20(a)] shows the optimal path for general pedestrians (obtained by applying the Dijkstra algorithm) and [Figure 4-20(b)] shows the optimal route for PWMD by applying the Dijkstra algorithm, after extracting the feasible graph (applying ONALIN). As mentioned above, travel distance was used as a cost, and the 'inaccessible' property was used when extracting feasible networks for PWMD.



(a) Optimal route graph for general pedestrians



(b) Optimal route graph for PWMD

[Figure 4-20] The result of multi-floor routing

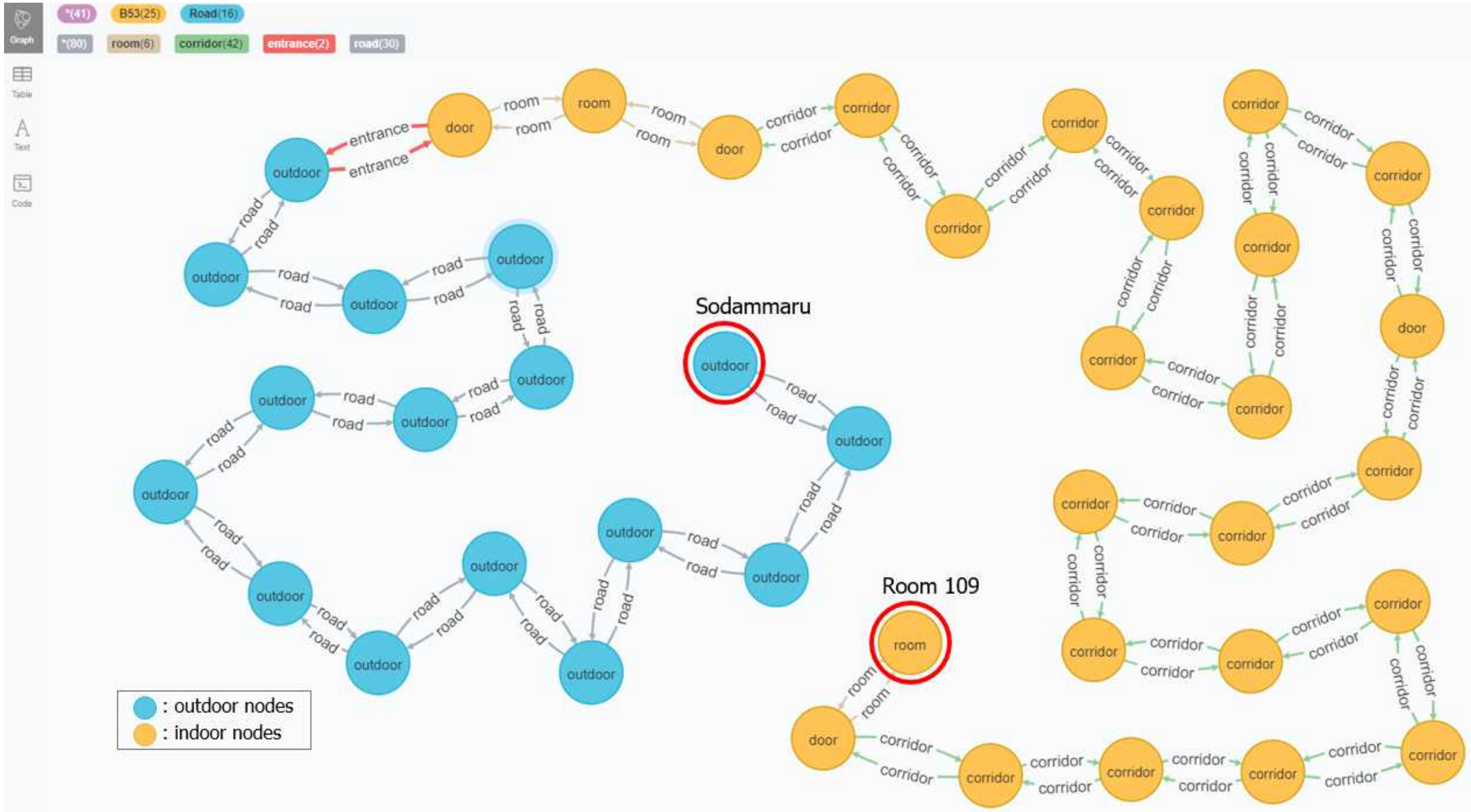
[Table 4-12] shows the number of nodes and estimated horizontal travel lengths of the two optimal routes. Compared to the PWMD routing, which required only feasible graphs, the optimal path for general pedestrians had a shorter distance cost and fewer corridor nodes, as all nodes and relationships could be included in the routes. The significant difference between the two paths was for vertical movement as shown with thicker arrows in [Figure 4-20]. For general pedestrians, the vertical transition from the fifth to the second floor was achieved using the nearest stair. However, the optimal route for PWMD only included a vertical transition through the elevator. This shows that the paths were planned appropriately, reflecting the accessibility of the spaces and facilities regarding PWMD. In conclusion, the indoor graph database generated through the proposed schema was shown to provide meaningful information on features that affect the mobility of PWMD. Furthermore, it is confirmed that the generated graph database can also cover the functionality of existing network model for supporting spatial routing.

[Table 4-12] Quantitative result for multi-floor routing

Route type	The number of nodes			Estimated horizontal travel length(m)	Vertical Movement Type
	Door	Room	Corridor		
Pedestrian	7	5	23	105.03	Both stair and elevator
PWMD	7	5	26	115.26	Elevator

Case 2. Integrated indoor-outdoor routing (space-seamless routing)

Integrated indoor-outdoor routing was performed to evaluate the utility of the indoor graph database. The scenario included travel starting from lecture room 109 on the first floor of Building 53 (Set-C) and arriving at the Sodammaru Restaurant, which was located in another building ten minutes away from the origin. As this was an outdoor graph, the data of the test area were downloaded from OSM. The outdoor network was manually modified, and then converted into a graph database in the same way of proposed conversion procedure. However, it was assumed that all nodes and links of the outdoor graph were accessible, because there was no available accessibility information in the OSM data of the corresponding area. The main exit node was created in the indoor graph database of the test building and an “entrance” labeled relationship with the nearest outdoor node was created to connect the two individual data sources. Distance-based path planning was conducted in the same way as in Case 1, after extracting feasible graphs for the indoor environments. [Figure 4-21] shows the integrated indoor-outdoor routing results and [Table 4-13] provides the estimated travel length for the optimal route.

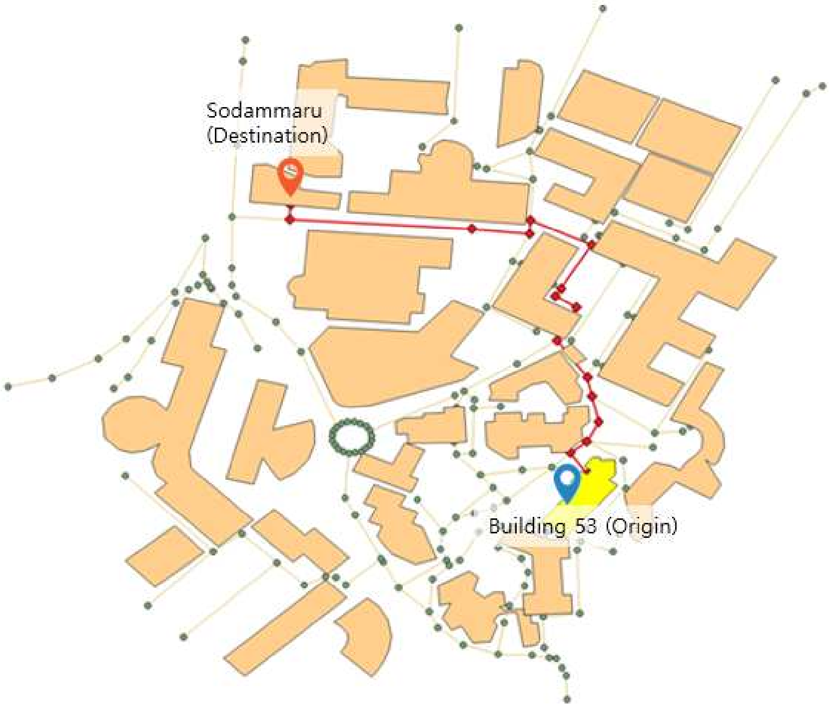


[Figure 4-21] The result of integrated indoor-outdoor routing

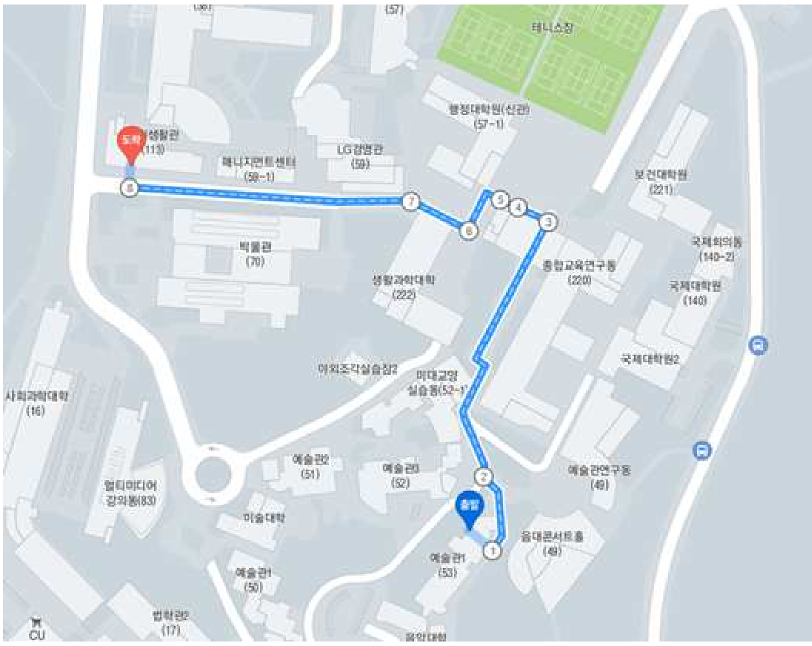
[Table 4-13] Quantitative result for integrated indoor-outdoor routing

Route type	The number of nodes			Estimated travel length(m)
indoor graph	Door	Room	Corridor	45.47
	4	2	19	
outdoor graph	16			446.34

The optimal outdoor route was compared with the walking path found by Kakao map. To allow for an intuitive comparison, the result of the graph database routing was visualized using the OSM network model ([Figure 4-22(a)]). There is a slight difference in the former route, including the front road of the building, and the latter route, including the back road of the building, but this is considered to be because the front road, included in the OSM network model, has not been established in the Kakao map's network model. Consquently, the optimal outdoor route using the graph database was almost identical to the optimal path from Kakao navigation ([Figure 4-22]). In other words, the identical result from the distance-based spatial routing using the existing database could also be obtained by using the graph database. Therefore, it is expected that the graph database can be an alternative for spatial routing.



(a) Outdoor route created using OSM data



(b) Outdoor route obtained using Kakao map

[Figure 4-22] Comparison of the outdoor route using OSM data and the route in Kakao map

Many complicated requirements are necessary to perform integrated indoor-outdoor routing using the network model. Generally, indoor data are constructed using a relative coordinate system, unlike outdoor data. In particular, when an indoor network model is created based on floor plans, as in this study, the network model's scale is determined according to the floor plan scale. The two data must be connected using an anchor node to perform routing by combining indoor and outdoor environments (Kim *et al.*, 2019; Li *et al.*, 2019; Wagner *et al.*, 2017). Thus, transformations of the scale and the coordinate system are required. Moreover, the schema needs to be unified to integrate two independently constructed data sources. The indoor graph database has a flexible schema and scale, however, which means that multiple independently generated data can be combined simply by creating relationships for the connection. The second routing case proves the high utility of the graph database in this regard. The outdoor network downloaded from the OSM and the indoor graph database generated using floor plans through the proposed method could be connected simply by creating an "entrance" relationship, without the need for additional integration work. Furthermore, the two independent graphs were successfully connected and used for seamless path-finding within a short time (within 10ms).

5. Conclusion

It is difficult for PWMD to travel freely indoors because of various restrictions, thus the mobility of PWMD is a significant social issue. The interest in indoor navigation services, especially regarding services that provide information on various obstacles and constraints, is increasing in accordance with recent improvements in indoor positioning technologies. This interest has led to demands for not only distance-based spatial routing, but also routing that considers the preferences of target users with special needs. Therefore, it is essential to generate a flexible database that can cope with this increased diversity of routing. Moreover, the corresponding database should also be based on a model containing sufficient PWMD-related information in order to be used in such services.

In this study, a technique for generating an indoor graph database using scanned floor plans was proposed. First, the data model was developed by deriving PWMD-related features and factors from relevant regulations. Also, the accessibility index for quantifying the difficulties in accessing spaces was designed based on the data model. Then, the process for generating a target database from scanned floor plans was proposed in three stages: 1) the indoor structure map is generated from scanned floor plans, 2) the indoor network model for PWMD is created by topology retrieval and accessibility assessment, 3) the network model is converted into an indoor graph database based on its connectivity. For generating an indoor structure map, the modified ResNet-based model, as a pre-trained model, was fine-tunned by the transfer learning-based approach.

The indoor graph database for PWMD navigation was generated for Seoul National University. Individual quantitative evaluations were conducted to intermediate outputs. Indoor structure maps showed an average pixel-wise accuracy of 87%, similar to the result of the pre-trained model. Indoor network models showed an average topological accuracy of 93%. The network models were connected horizontally and vertically. After generating the constraint information considering the mobility of PWMD using the proposed accessibility index, the final indoor graph database was completely generated. The appropriateness of the designed data model and the generated indoor graph database were verified through two scenario-based routings. The multi-floor routing test showed that proper route detouring inaccessible spaces was extracted using the generated indoor graph database. Consequently, it was confirmed that the indoor graph database generated through the proposed technique adequately described the indoor environment with relevant information in terms of PWMD. In addition, it was verified the generated graph database can support spatial routing like a network model does. In the integrated indoor-outdoor routing test, usability of generated graph database was proved through seamless path planning from indoor to outdoor without a complicated data integration process.

The results of this study are concluded as follows. The indoor graph database suitable for PWMD navigation can be generated using scanned floor plans through the proposed process. The data model, which represents the indoor environment in terms of PWMD, was developed to generate an appropriate indoor database for PWMD. Also, sub-procedures for retrieving information were proposed to generate a target database using scanned floor plans by the end-to-end process. The target database can be easily shared since the developed data model for the target database is based on IndoorGML, the standard of indoor spatial information. Also, as scanned floor plans can be easily acquired and sub-procedures of the entire process were automated, the required time and cost for data generation will be reduced. An indoor database generated by the proposed process is easily integrated with various types of information due to the characteristics of the graph database. Thus, an indoor graph database is expected to be utilized for diversified routing, such as personalized navigation. Moreover, the proposed technique could also be applied to pre-built spatial networks because a network-to-graph database conversion is included in the technique.

The optimal trained model for generating structure maps in this study may have limitations regarding the input floor plan format. In other words, additional learning is required when inputting floor plans that are drawn in different fashions. Nevertheless, it is expected that only a small additional training cost would be necessary, as the proposed technique does not require a large amount of training data, nor does it need a complex learning process. Furthermore, although the data accuracy decreases for the unique-shaped building, the proposed technique is effective enough to build initial data, which could then be adjusted through manual editing. The suggested approach generates intermediate outputs such as indoor structure maps and indoor network models. Therefore, the final indoor graph database is dependent on the results of the preceding steps. Accordingly, further research to improve these subdivided procedures will be conducted to increase the accuracy of the results generated in each step.

This study is cornerstone research that attempted to utilize a graph database for a spatial application. Its scope is limited to generating a graph database focusing on geometry and topology for PWMD navigation services. Research into methods to generate a multi-layered indoor graph database for personalized routing, integrating additional information such as POI, unstructured textual data, and existing spatial information, will be performed in the future.

References

Al Delail, B., Weruaga, L., & Zemerly, M. J. (2012). CAViAR: Context aware visual indoor augmented reality for a university campus. In *Proceedings of the 2012 IEEE/WIC/ACM International Conferences on Web Intelligence and Intelligent Agent Technology.*

Alqahtani, E. J., Alshamrani, F. H., Syed, H. F., & Alhaidari, F. A. (2018). Survey on algorithms and techniques for indoor navigation systems. In *Proceedings of the 2018 21st Saudi Computer Society National Computer Conference*.

Bendel, J., & Klüpfel, H. (2011). Accessibility and evacuation planning – Similarities and differences. *Pedestrian and evacuation dynamics,* 701–712.

Bian, W., Guo, Y., & Qiu, Q. (2014). Research on personalized indoor routing algorithm. In *Proceedings of the 2014 13th International Symposium on Distributed Computing and Applications to Business, Engineering and Science*.

Burghardt, D., & Steiniger, S. (2005). Usage of principal component analysis in the process of automated generalisation. In *Proceedings of the 22nd international cartographic conference*.

Castelluccio, M., Poggi, G., Sansone, C., & Verdoliva, L. (2015). Land use classification in remote sensing images by convolutional neural networks. *arXiv preprint arXiv:1508.00092.*

Cheraghi, S. A., Sharma, A., Namboodiri, V., & Arsal, G. (2019). SafeExit4AII: an inclusive indoor emergency evacuation system for people with disabilities. In *Proceedings of the 16th Web For All 2019 Personalization-Personalizing the Web*.

Church, R. L., & Marston, J. R. (2003). Measuring accessibility for people with a disability. *Geographical Analysis*, *35*(1), 83-96.

Clementini, E., & Pagliaro, A. (2020). The construction of a network for indoor navigation. In *Proceedings of the GISTAM*.

Costa, C., Ge, X., & Chrysanthis, P. (2019). CAPRIO: Context-aware path recommendation exploiting indoor and outdoor information. In *Proceedings of the 20th IEEE International Conference on Mobile Data Management*.

Dao, T. H. D., & Thill, J. C. (2018). Three dimensional indoor network accessibility auditing for floor plan design. *Transactions in GIS*, *22*(1), 288-310.

Davis, J., & Goadrich, M. (2006). The relationship between Precision-Recall and ROC curves. In *Proceedings of the 23rd international conference on Machine learning*.

De Sousa, V. M., & Cura, L. M. D. V. (2018). Logical design of graph databases from an entity-relationship conceptual model. In *Proceedings of the 20th International Conference on Information Integration and Web-based Applications & Services*.

Dilts, T. (2015). Polygon to centerline tool for arcgis. *University of*

Nevada Reno: Reno, NV, USA.

Dodge, S., Xu, J., & Stenger, B. (2017). Parsing floor plan images. In *Proceedings of the 2017 Fifteenth IAPR international conference on machine vision applications*.

Dudas, P. M., Ghafourian, M., & Karimi, H. A. (2009). ONALIN: Ontology and algorithm for indoor routing. In *Proceedings of the 2009 Tenth International Conference on Mobile Data Management: Systems, Services and Middleware*.

Evcil, A. N. (2012). Raising awareness about accessibility. *Procedia-Social and Behavioral Sciences, 47*, 490-494.

Fernandes, D., & Bernardino, J. (2018). Graph databases comparison: AllegroGraph, ArangoDB, InfiniteGraph, Neo4J, and OrientDB. *DATA,* 373-380.

Fleiner, R., Szász, B., Simon-Nagy, G., & Micsik, A. (2017). Indoor navigation for motion disabled persons in medical facilities. *Acta Polytechnica Hungarica, 14*(1), 111-128.

Fu, M., Liu, R., Qi, B., & Issa, R. R. (2020). Generating straight skeleton-based navigation networks with Industry Foundation Classes for indoor way-finding. *Automation in Construction, 112*, 103057.

Ge, H., Bessho, M., Koshizuka, N., & Sakamura, K. (2015). Model design of generating path with accessibility semantics for assisting indoor mobility. In *Proceedings of the 2015 IEEE International Conference on Data Science and Data Intensive*

Systems.

Goetz, M. (2012). Using crowdsourced indoor geodata for the creation of a three-dimensional indoor routing web application. *Future Internet, 4*(2), 575-591.

Goetz, M., & Zipf, A. (2011). Formal definition of a user-adaptive and length-optimal routing graph for complex indoor environments. *Geo-Spatial Information Science, 14*(2), 119-128.

Gu, Y., Chen, Y., Jiang, X., Wang, J., Xiao, Y., & Wu, T. (2019). CoFINLo: Coarse to fine indoor navigation and localization system. In *Proceedings of the 2019 IEEE SmartWorld, Ubiquitous Intelligence & Computing, Advanced & Trusted Computing, Scalable Computing & Communications, Cloud & Big Data Computing, Internet of People and Smart City Innovation*.

Guo, S., Niu, G., Wang, Z., Pun, M. O., & Yang, K. (2020). An indoor knowledge graph framework for efficient pedestrian localization. *IEEE Sensors Journal, 21*(4), 5151-5163.

Department of Justice. (2010). ADA standards for accessible design. USA.

Hamieh, A., Makhlouf, A. B., Louhichi, B., & Deneux, D. (2020). A BIM-based method to plan indoor paths. *Automation in Construction, 113*, 103120.

Hashemi, M. (2018). Dynamic, stream-balancing, turn-minimizing, accessible wayfinding for emergency evacuation of people who use a wheelchair. *Fire technology, 54*(5), 1195-1217.

Hashemi, M., & Karimi, H. A. (2016). Indoor spatial model and accessibility index for emergency evacuation of people with disabilities. *Journal of Computing in Civil Engineering, 30*(4), 04015056.

Hor, A. E., Gunho, S., Claudio, P., Jadidi, M., & Afnan, A. (2018). A semantic graph database for BIM-GIS integrated information model for an intelligent urban mobility web application. *ISPRS Annals of Photogrammetry, Remote Sensing & Spatial Information Sciences, 4*(4), 89-96.

Iida, H., Hiroi, K. E. I., Kaji, K., & Kawaguchi, N. (2015). A proposal of IndoorGML extended data model for pedestrian-oriented voice navigation system. In *Proceedings of the Seventh ACM SIGSPATIAL International Workshop on Indoor Spatial Awareness*.

Isikdag, U., Zlatanova, S., & Underwood, J. (2013). A BIM-Oriented Model for supporting indoor navigation requirements. *Computers, Environment and Urban Systems, 41*, 112-123.

Ismail, A., Nahar, A., & Scherer, R. (2017). Application of graph databases and graph theory concepts for advanced analysing of BIM models based on IFC standard. In *Proceedings of the European Group for Intelligent Computing in Engineering.*

Ismail, A., Strug, B., & Ślusarczyk, G. (2018). Building knowledge extraction from BIM/IFC data for analysis in graph databases. In *Proceedings of the International Conference on Artificial*

Intelligence and Soft Computing.

Isola, P., Zhu, J. Y., Zhou, T., & Efros, A. A. (2017). Image-to-image translation with conditional adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*.

Jabbar, J. A., & Bulbul, R. (2019). Semantic enrichment of routing engines using linked data: A case study using graphhopper. *International Archives of the Photogrammetry, Remote Sensing & Spatial Information Sciences, 42*(4/W14), 111-117.

Jaiswal, G., & Agrawal, A. P. (2013). Comparative analysis of Relational and Graph databases. *IOSR Journal of Engineering, 3*(8), 2250-3021.

Jang, H., Yu, K., & Yang, J. (2020). Indoor reconstruction from floorplan images with a deep learning approach. *ISPRS International Journal of Geo-Information*, *9*(2), 65.

Jung, H. J., & Lee, J. (2017). Development of an omnidirectional-image-based data model through extending the IndoorGML concept to an indoor patrol service. *Journal of Sensors*, 1-15.

Kang, H. K., & Li, K. J. (2017). A standard indoor spatial data model —OGC IndoorGML and implementation approaches. *ISPRS International Journal of Geo-Information, 6*(4), 116.

Karas, I. R., Batuk, F., Akay, A. E., & Baz, I. (2006). Automatically extracting 3D models and network analysis for indoors.

Innovations in 3D Geo Information Systems, 395-404.

Karimi, H. A., & Ghafourian, M. (2010). Indoor routing for individuals with special needs and preferences. *Transactions in GIS, 14*(3), 299-329.

Karimi, H. A., Zhang, L., & Benner, J. G. (2014). Personalized accessibility map (PAM): A novel assisted wayfinding approach for people with disabilities. *Annals of GIS, 20*(2), 99-108.

Khalili, A., & Chua, D. H. (2015). IFC-based graph data model for topological queries on building elements. *Journal of Computing in Civil Engineering, 29*(3), 04014046.

Kim, B. G., Li, K. J., & Kang, H. K. (2015). Generation of indoor network by crowdsourcing. *Spatial Information Research*, *23*(1), 49-57.

Kim, J. (2014). *Automatic generation of pedestrian level network data* (Doctoral dissertation). Seoul National University, Korea.

Kim, J. (2020). *Floorplan analysis method for large and complicated buildings using geometric extraction and vectorization-case study on Seoul National University building floorplan-* (Masters dissertation). Seoul National University, Korea.

Kim, J., Huh, Y., Kim, D. S., Yu, K. (2011). A new method for automatic areal feature matching based on shape similarity using CRITIC method. *Journal of the Korean Society of Surveying, Geodesy, Photogrammetry and Cartography, 29*(2), 113-121.

Kim, S. Y., Park, Y. Y., Lee, K. O., Jung, J. B., & Kye, J. E. (2021). A study on the making indoor topology map for the autonomous driving of wearable robots. *Elementary Education Online, 20*(3), 1153-1159.

Kim, S., Park, S., Kim, H., & Yu, K. (2021). Deep floor plan analysis for complicated drawings based on style transfer. *Journal of Computing in Civil Engineering, 35*(2), 04020066.

Kim, T., Kim, K. S., & Lee, J. (2019). How to extend IndoorGML for seamless navigation between indoor and outdoor space. In *Proceedings of the International Symposium on Web and Wireless Geographical Information Systems*.

Kim, Y., Kang, H., & Lee, J. (2014). Developing CityGML indoor ADE to manage indoor facilities. *Innovations in 3D Geo-information sciences,* 243-265.

Korea Parliament (2020). Act on Promotion of the Transportation Convenience of Mobility Disadvantaged Persons (Act No. 17545, 2020. 10. 20, Partial Amendment. Enforcement Date, 21, April, 2021).

Kostic, N., & Scheider, S. (2015). Automated generation of indoor accessibility information for mobility-impaired individuals. In *Proceedings of the AGILE 2015*.

Lee, K. (2012). A review on the level comparison of barrier-free certification criteria. *Journal of Rehabilitation Research*, *16*, 159-187.

Lee, J. (2004). A spatial access-oriented implementation of a 3-D GIS topological data model for urban entities. *GeoInformatica, 8*(3), 237-264.

Lee, H., Choi, C., & Lee, T. (2010). A comparative study on the barrier free dwelling guidelines of domestic and foreign countries, *Journal of The Korea Institute of Healthcare Architecture, 16*(1), 65-75.

Lewandowicz, E., Lisowski, P., & Flisek, P. (2019). A modified methodology for generating indoor navigation models. *ISPRS International Journal of Geo-Information, 8*(2), 60.

Li, K. J., Conti, G., Konstantinidis, E., Zlatanova, S., & Bamidis, P. (2019). OGC IndoorGML: A standard approach for indoor maps. *Geographical and Fingerprinting Data to Create Systems for Indoor Positioning and Indoor/Outdoor Navigation,* 187-207.

Li, K. J., & Lee, J. Y. (2013). Basic concepts of indoor spatial information candidate standard IndoorGML and its applications. *Spatial Information Research, 21*(3), 1-10.

Li, W., Hu, D., & Lin, Z. (2018). Indoor space dimensional model supporting the barrier-free path-finding. In *Proceedings of the 2018 Ubiquitous Positioning, Indoor Navigation and Location-Based Services*.

Liu, C., Wu, J., Kohli, P., & Furukawa, Y. (2017). Raster-to-vector: revisiting floorplan transformation. In *Proceedings of the IEEE*

International Conference on Computer Vision.

Liu, L. (2017). Indoor semantic modelling for routing: The two-level routing approach for indoor navigation. *A+ BE| Architecture and the Built Environment*, (17), 1-252.

Liu, L., & Zlatanova, S. (2012). A semantic data model for indoor navigation. In *Proceedings of the Fourth ACM SIGSPATIAL International Workshop on Indoor Spatial Awareness*.

Liu, L., & Zlatanova, S. (2013). A two-level path-finding strategy for indoor navigation. *Intelligent systems for crisis management,* 31-42.

Liu, L., Zlatanova, S., Li, B., van Oosterom, P., Liu, H., & Barton, J. (2019). Indoor routing on logical network using space semantics. *ISPRS International Journal of Geo-Information, 8*(3), 126.

Maheshwari, N., Srivastava, S., & Rajan, K. S. (2019). Development of an indoor space semantic model and its implementation as an IndoorGML extension. *ISPRS International Journal of Geo-Information, 8*(8), 333.

Meijers, M., Zlatanova, S., & Pfeifer, N. (2005). 3D geo-information indoors: structuring for evacuation. In *Proceedings of the workshop on next generation 3D city models*.

Ministry of Health and Welfare (2015). Regulations for certification of living environment without obstacles. Available online: https://bf.koddi.or.kr/ (accessed on 06 April 2021)

Ministry of Land, Infrastructure and Transport (2020). Developing world-leading cost effective indoor spatial information core technologies and its test beds.

Mirahadi, F., & McCabe, B. Y. (2021). EvacuSafe: A real-time model for building evacuation based on Dijkstra's algorithm. *Journal of Building Engineering*, *34*, 101687.

Mortari, F., Clementini, E., Zlatanova, S., & Liu, L. (2019). An indoor navigation model and its network extraction. *Applied Geomatics, 11*(4), 413-427.

Niu, L., & Song, Y. Q. (2019). A faster R-CNN approach for extracting indoor navigation graph from building designs. In *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, 42*(2/W13), 865-672.

OGC. (2020). OGC IndoorGML 1.1, Document No. 19-011r4. Available online: http://www.opengeospatial.org/standards/indoorgml (accessed on 06 April 2021).

Onorati, T., Malizia, A., Diaz, P., & Aedo, I. (2014). Modeling an ontology on accessible evacuation routes for emergencies. *Expert systems with Applications, 41*(16), 7124-7134.

Pan, S. J., & Yang, Q. (2009). A survey on transfer learning. *IEEE Transactions on knowledge and data engineering, 22*(10), 1345-1359.

Pang, G. K., Takahashi, K., Yokota, T., & Takenaga, H. (2002). Intelligent route selection for in-vehicle navigation systems.

Transportation Planning and Technology, 25(3), 175-213.

Park, S., & Yu, K. (2019). Automatic floor matching for 3D indoor spatial modeling. In *Proceedings of the 15th International Conference on Location-Based Services*.

Park, S., Yu, K., & Kim, J. (2020). Data model for IndoorGML extension to support indoor navigation of people with mobility disabilities. I*SPRS International Journal of Geo-Information, 9*(2), 66.

Peeta, S., & Yu, J. W. (2005). A hybrid model for driver route choice incorporating en-route attributes and real-time information effects. *Networks and Spatial Economics, 5*(1), 21-40.

Qin, H., Rice, R. M., Fuhrmann, S., Rice, M. T., Curtin, K. M., & Ong, E. (2016). Geocrowdsourcing and accessibility for dynamic environments. *GeoJournal, 81*(5), 699-716.

Rimmer, J. H., Riley, B., Wang, E., & Rauworth, A. (2005). Accessibility of health clubs for people with mobility disabilities and visual impairments. *American journal of public health*, *95*(11), 2022-2028.

Roy-Hubara, N., Rokach, L., Shapira, B., & Shoval, P. (2017). Modeling graph database schema. *IT Professional, 19*(6), 34-43.

Sabri, Q. U., Bayer, J., Ayzenshtadt, V., Bukhari, S. S., Althoff, K. D., & Dengel, A. (2017). Semantic pattern-based retrieval of architectural floor plans with case-based and graph-based searching techniques and their evaluation and visualization. In *Proceedings of the International Conference on Pattern Recognition Applications and Methods*.

Song, Y., Niu, L., Liu, P., & Li, Y. (2021). Fire hazard assessment with indoor spaces for evacuation route selection in building fire scenarios. *Indoor and Built Environment*. 1-14.

Srivastavaa, S., Maheshwarib, N., & Rajanc, K. (2018). Towards generating semantically-rich IndoorGML data from architectural plans. In *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, 42*(4), 591-595.

Steinmetz, D., Dyballa, D., Ma, H., & Hartmann, S. (2018). Using a conceptual model to transform road networks from OpenStreetMap to a graph database. In *Proceedings of the International Conference on Conceptual Modeling*.

Suh, H., Kim, S., & Yu, K. (2020). CNN-based floor plan analysis using patch division. In *Proceedings of the Korean Society for Geospatial Information Science.*

Tan, C., Sun, F., Kong, T., Zhang, W., Yang, C., & Liu, C. (2018). A survey on deep transfer learning. In *Proceedings of the International conference on artificial neural networks*.

Taneja, S., Akinci, B., Garrett, J. H., Soibelman, L., & East, B. (2011). Transforming IFC-based building layout information into a geometric topology network for indoor navigation assistance. *Computing in Civil Engineering, 2011,* 315-322.

Taneja, S., Akinci, B., Garrett Jr, J. H., & Soibelman, L. (2016). Algorithms for automated generation of navigation models from building information models to support indoor map-matching. *Automation in Construction, 61*, 24-41.

Teo, T. A., & Cho, K. H. (2016). BIM-oriented indoor network model for indoor and outdoor combined route planning. *Advanced Engineering Informatics, 30*(3), 268-282.

The UN Refugee Agency (UNHCR). (2019). Working with Persons with Disabilities in Forced Displacement.

Vanclooster, A., Neutens, T., Fack, V., Van de Weghe, N., & De Maeyer, P. (2012). Measuring the exitability of buildings: A new perspective on indoor accessibility. *Applied Geography, 34*, 507-518.

Wagner, S., Fet, N., Handte, M., & Marrón, P. J. (2017). An approach for hybrid indoor/outdoor navigation. In *Proceedings of the 2017 International Conference on Intelligent Environments*.

Wang, Z., & Niu, L. (2018). A data model for using OpenStreetMap to integrate indoor and outdoor route planning. *Sensors, 18*(7), 2100.

Wiedemann, C. (2003). External evaluation of road networks. *International Archives of Photogrammetry Remote Sensing and Spatial Information Sciences, 34*(3/W8), 93-98.

Winter, S., Hamzei, E., Van de Weghe, N., & Ooms, K. (2018). A graph representation for verbal indoor route descriptions. In

Proceedings of the German Conference on Spatial Cognition.

Wu, H., Chou, X., Li, G., & Shi, Y. (2018). Indoor space model based on knowledge graph. In *Proceedings of the 10th International Conference on Intelligent Human-Machine Systems and Cybernetics*.

Wu, Y., Shang, J., Chen, P., Zlatanova, S., Hu, X., & Zhou, Z. (2020). Indoor mapping and modeling by parsing floor plan images. *International Journal of Geographical Information Science,* 1-27.

Xu, S. (2018). *Investigation of graph-databases for storing and analyzing building models* (Masters dissertation). Technische Universität München, Germany.

Yamasaki, T., Zhang, J., & Takada, Y. (2018). Apartment structure estimation using fully convolutional networks and graph model. In *Proceedings of the 2018 ACM Workshop on Multimedia for Real Estate Tech*.

Yan, J., Zlatanova, S., & Diakite, A. A. (2020). Two new pedestrian navigation path options based on semi-indoor space. *ISPRS Annals of Photogrammetry, Remote Sensing & Spatial Information Sciences*, *6,* 175-182.

Yang, L., Bi, H., Yao, X., & Chen, W. (2020). An indoor spatial accessible area generation approach considering distance constraints. *Annals of GIS, 26*(1), 25-34.

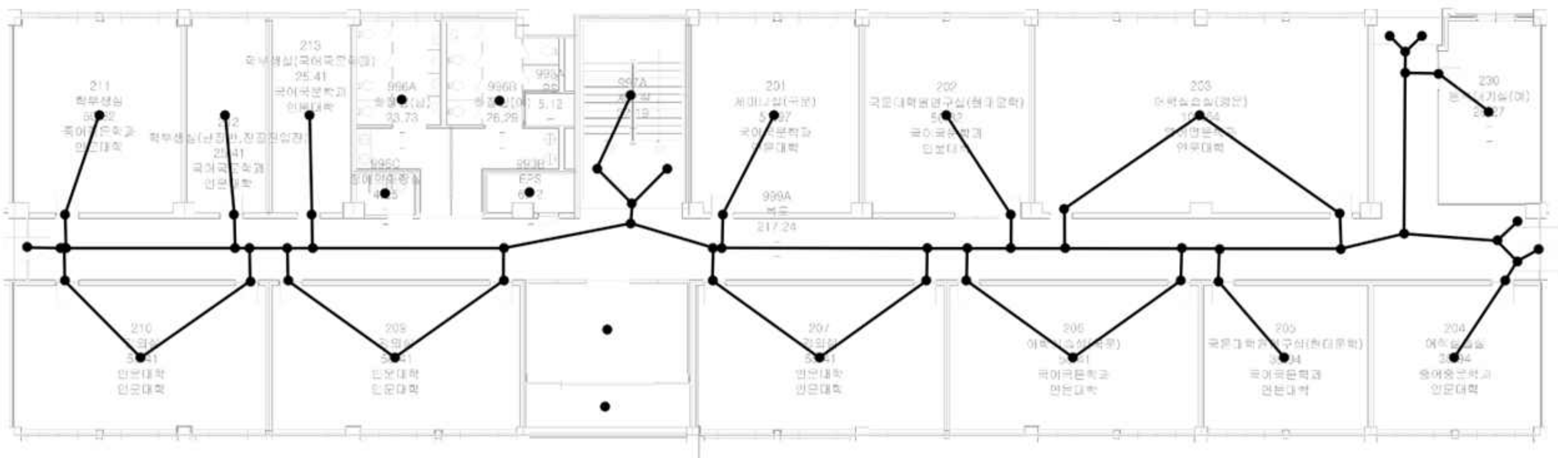
Yang, L., & Worboys, M. (2015). Generation of navigation graphs for indoor space. *International Journal of Geographical Information Science, 29*(10), 1737-1756.

Zeng, Z., Li, X., Yu, Y. K., & Fu, C. W. (2019). Deep floor plan recognition using a multi-task network with room-boundary-guided attention. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*.

Zhao, B., Huang, B., & Zhong, Y. (2017). Transfer learning with fully pretrained deep convolution networks for land-use classification. *IEEE Geoscience and Remote Sensing Letters, 14*(9), 1436-1440.

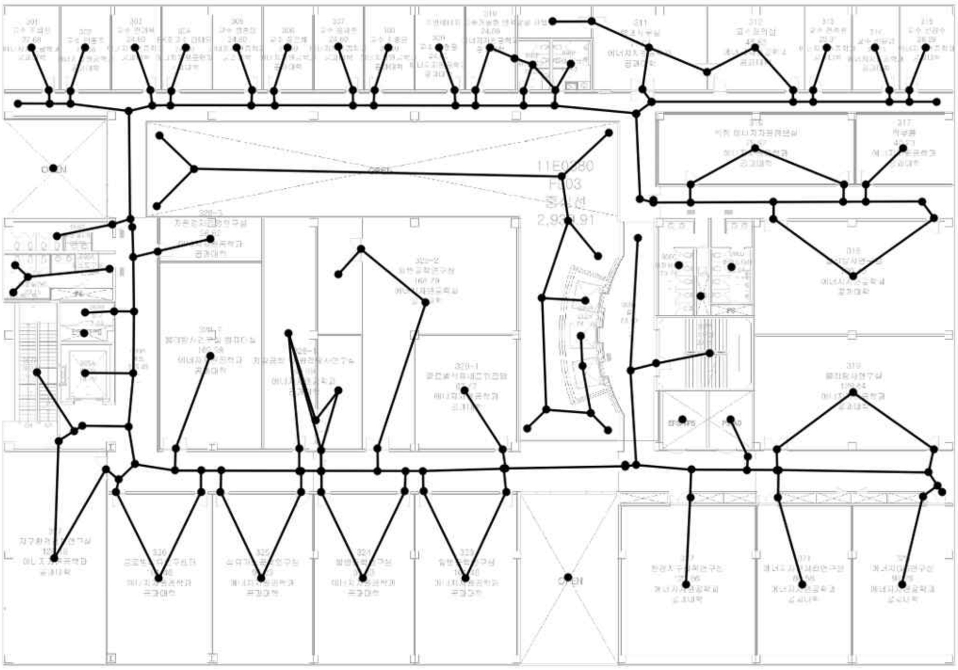
Appendix

A. Results of indoor network model



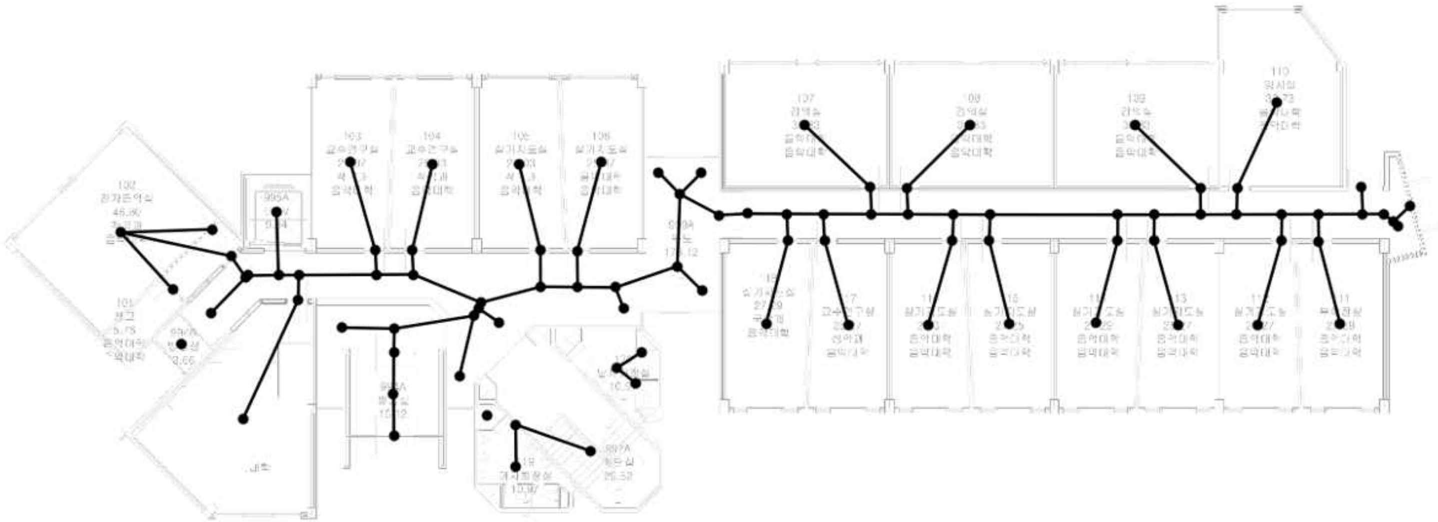
[Figure A-1] The created indoor network model for set-A

For the upper part, the stair was extracted as being located within the corridor instead of in an enclosed space, thus the 'staircase' link was connected to the nearest corridor wing edge. Also, the floor's main entrance (door) was connected to the endpoint of the corridor on the left-hand side.



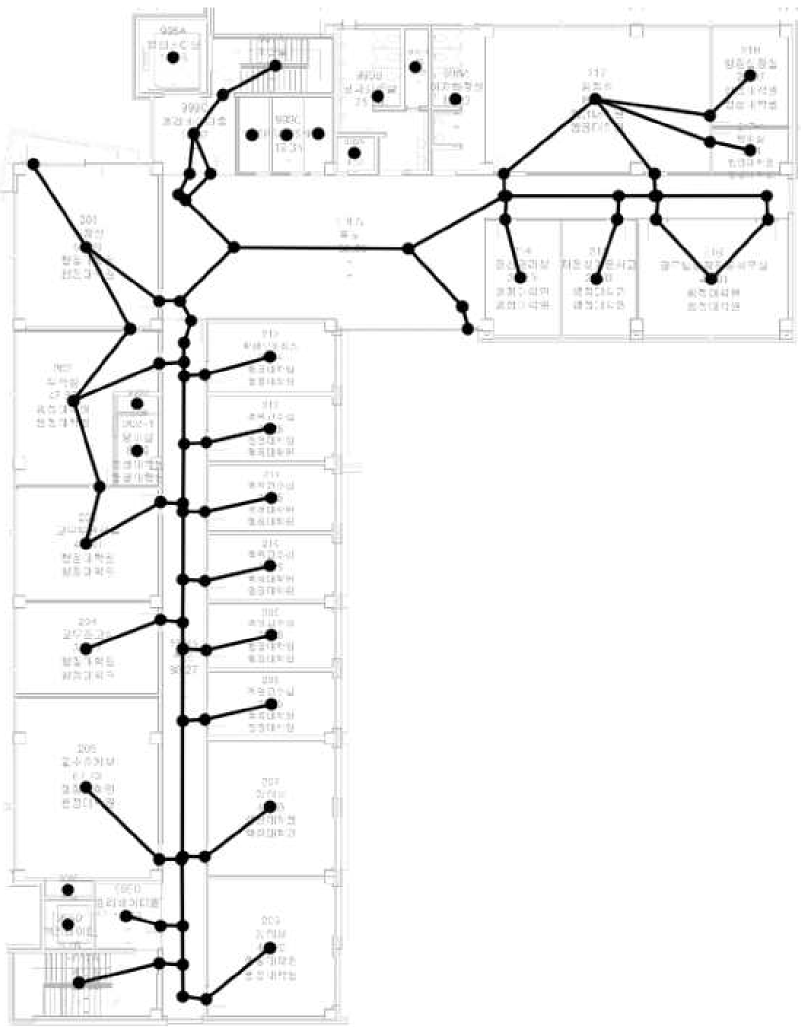
[Figure A-2] The created indoor network model for set-B

The space was not enclosed and merged into the corridor, thereby creating an irregular sub-network on the left-hand side. Also, it was combined with corridor through very narrow passage, said sub-network was isolated. Furthermore, the stairs were correctly extracted, but the door could not be joined to the associated staircase because the space including those stairs was not extracted as an enclosed room. Therefore, the stair node was connected to the nearest corridor. Three elevators were incorrectly connected to the corridor as the elevator door, which was not included in the annotation, could not be extracted as a feature in the indoor structure map.



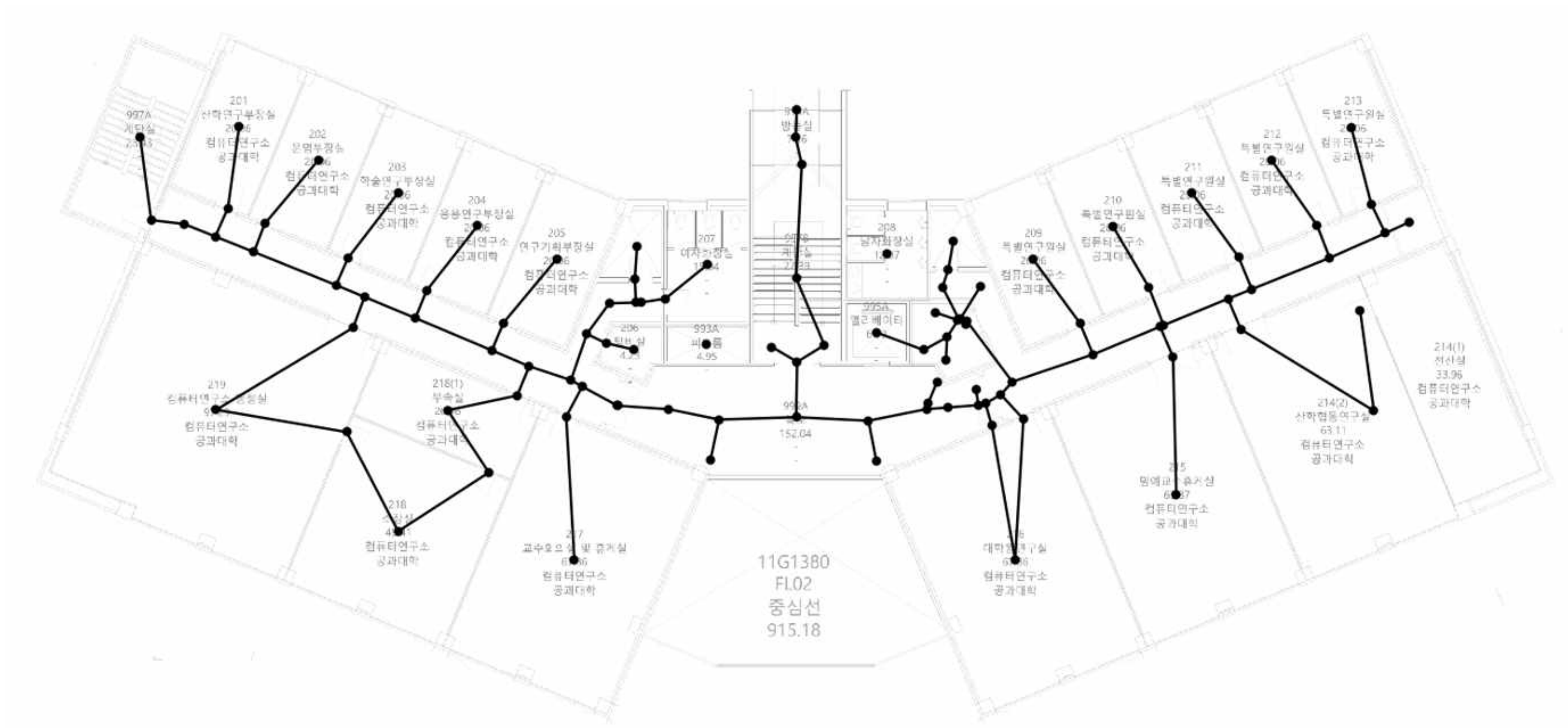
[Figure A-3] The created indoor network model for set-C

When connecting between corridors, the nearest neighbors (corridor nodes) were first searched and linked instead of the shortest connection, to prevent the increase of unnecessary nodes. Furthermore, the stair was not connected because it was extracted as an isolated space, without any associated door.



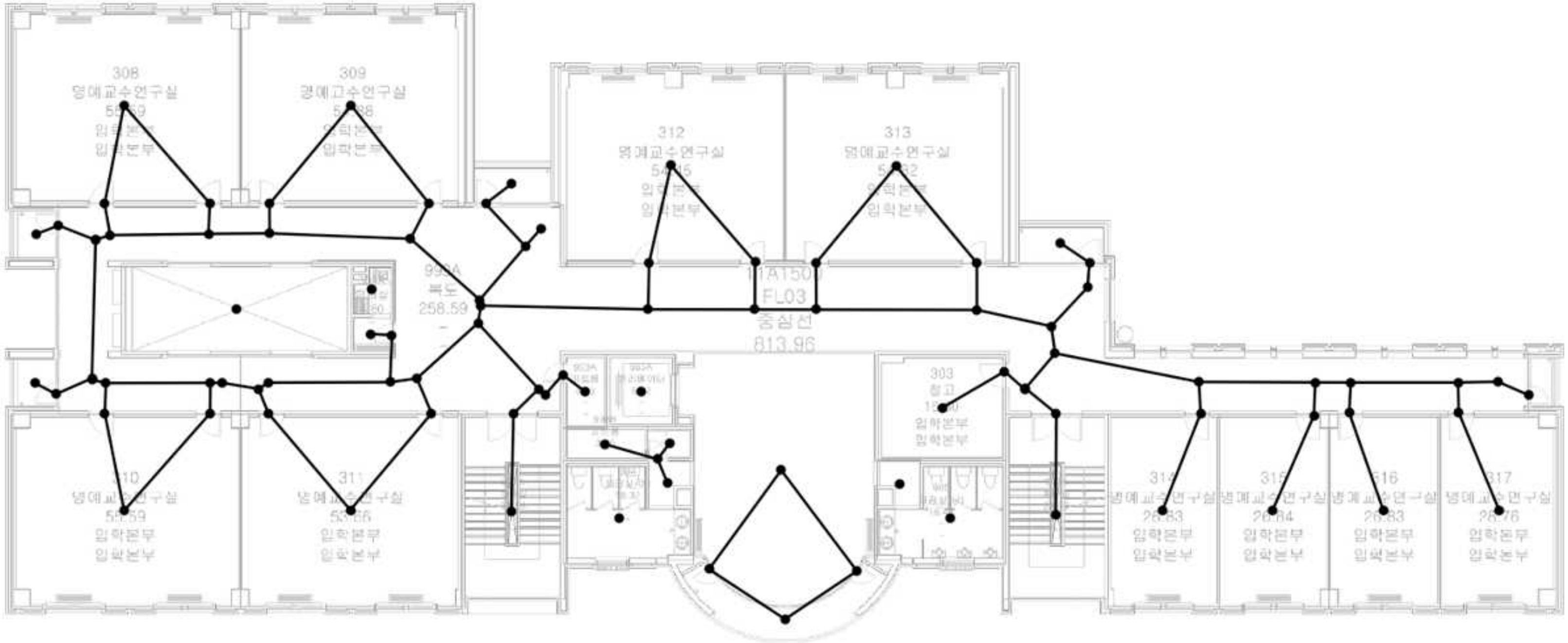
[Figure A-4] The created indoor network model for set-D

As the door symbols were not drawn in the floor plans, floating room nodes were created. Also, elevators could not be connected to the corridor because the elevator halls were extracted as isolated spaces without associated doors.



[Figure A-5] The created indoor network model for set-E

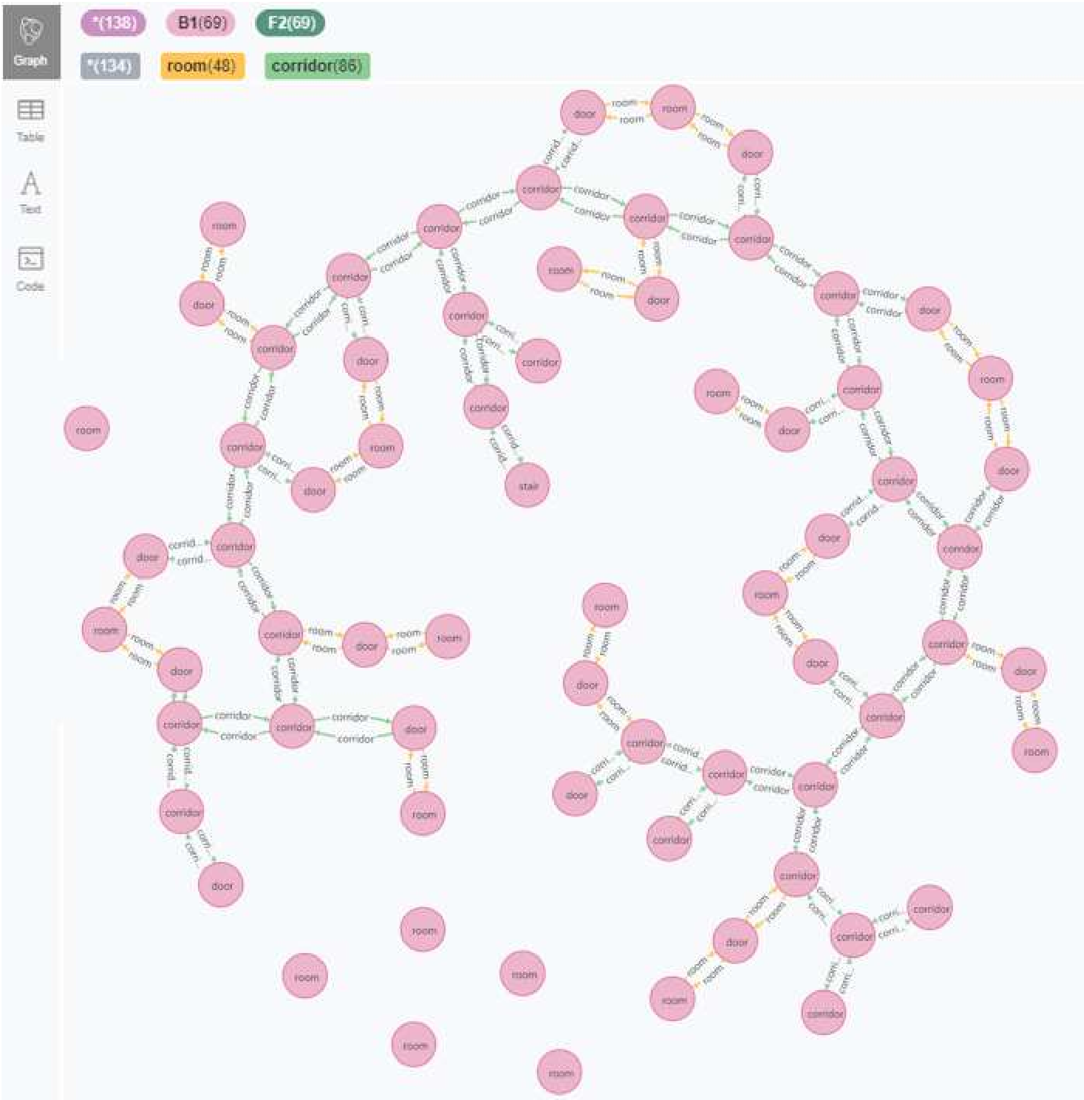
The stair was connected to one of the corridor wing edges since a stair within the corridor, not as a separate space. Also, as several complex, small areas were merged into the corridor, a detailed corridor network was created, and the number of corridor nodes increased. An unnecessary room link was created due to an incorrectly extracted door.



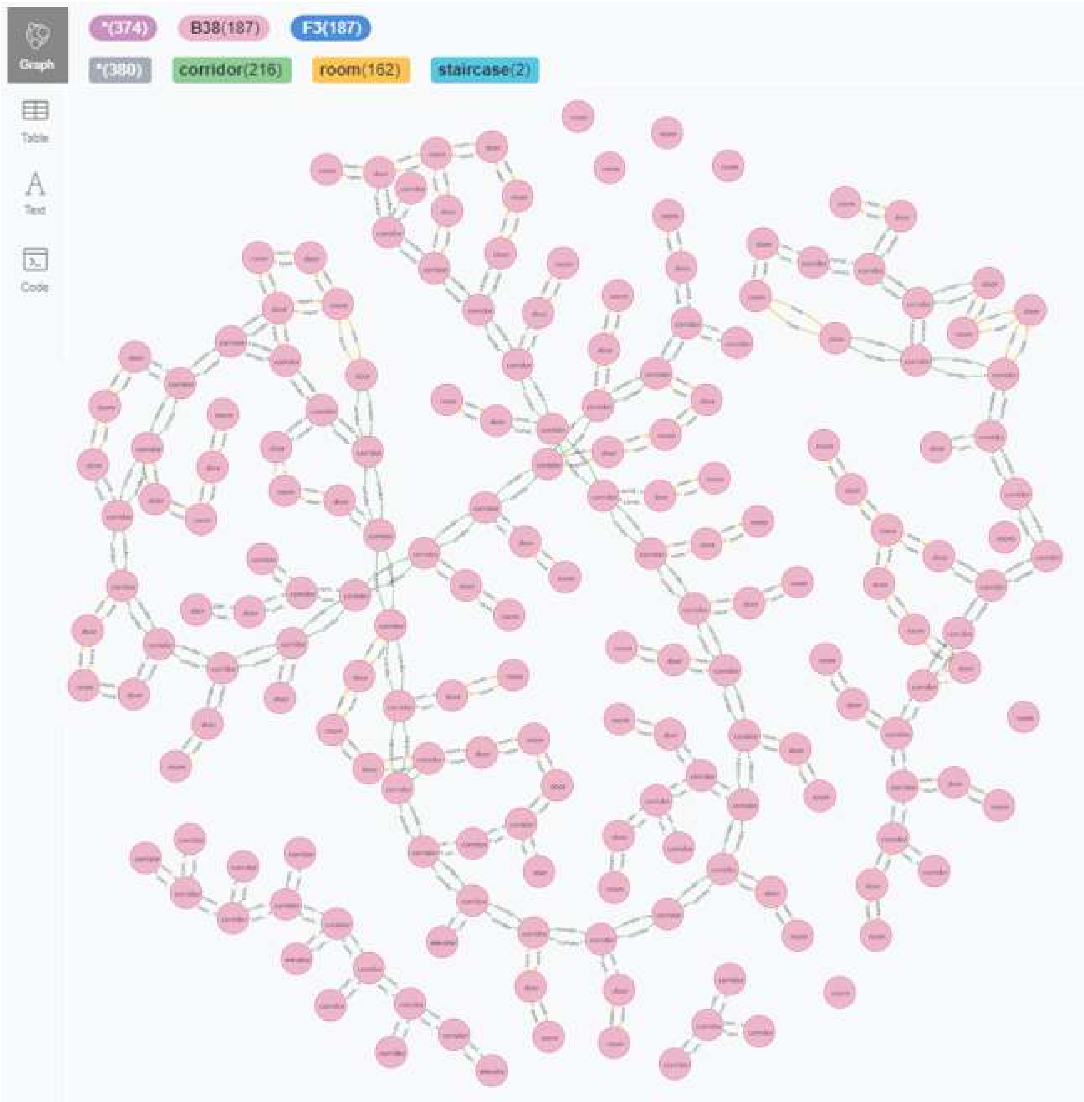
[Figure A-6] The created indoor network model for set-F

Disconnected networks were created because an isolated room without an associated door was extracted.

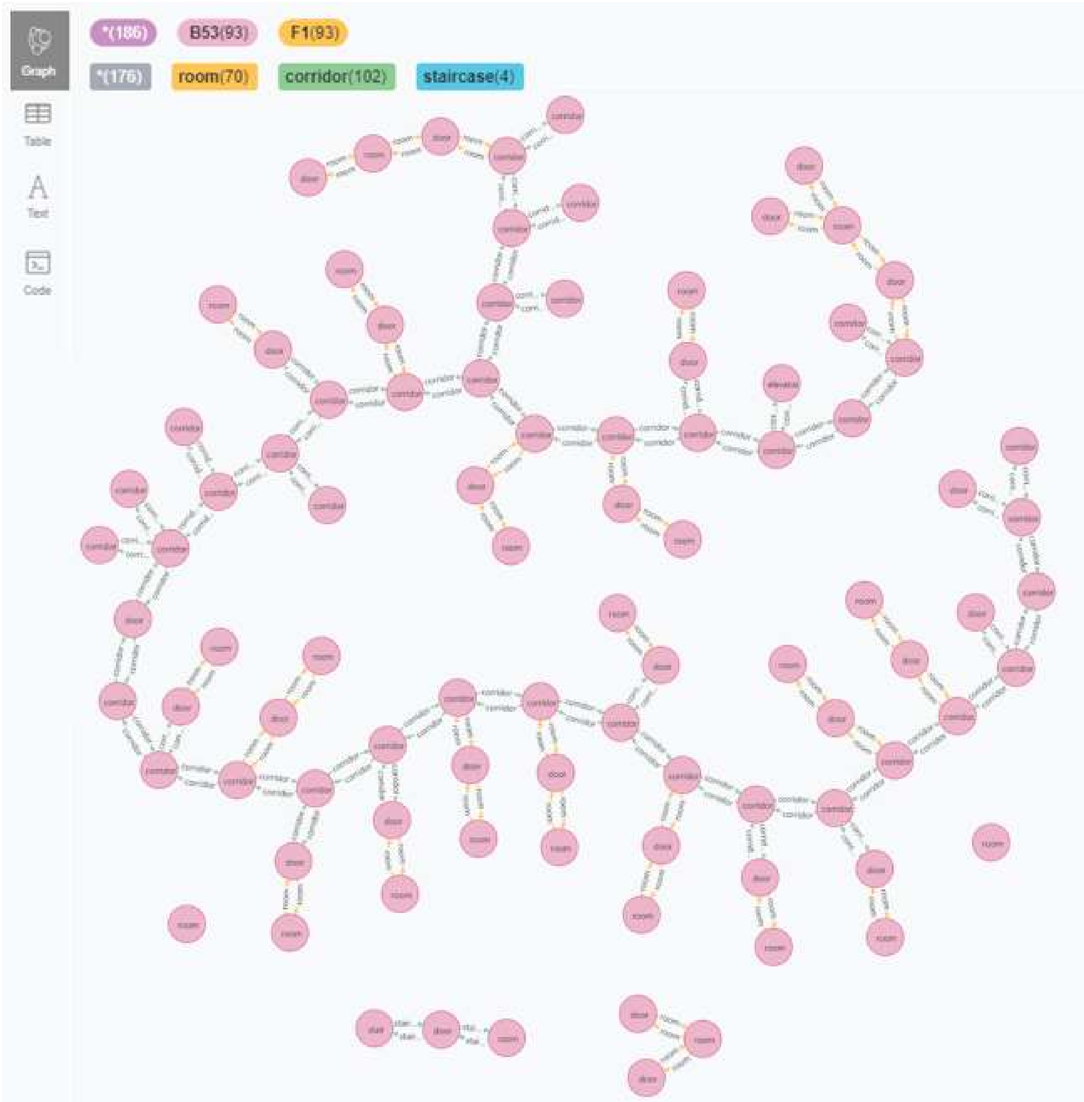
B. Results of indoor graph database



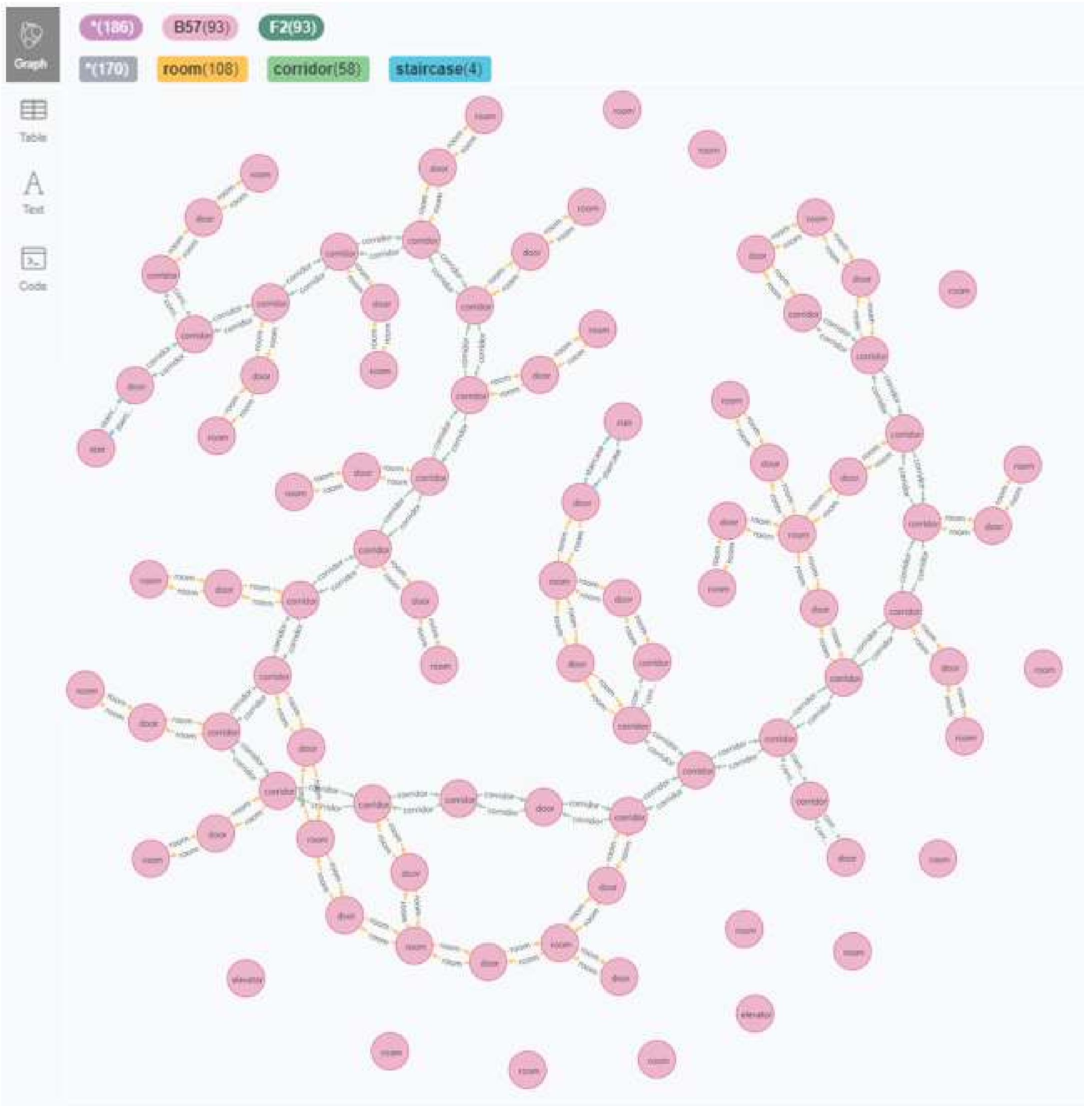
[Figure B-1] The generated indoor graph database for set-A



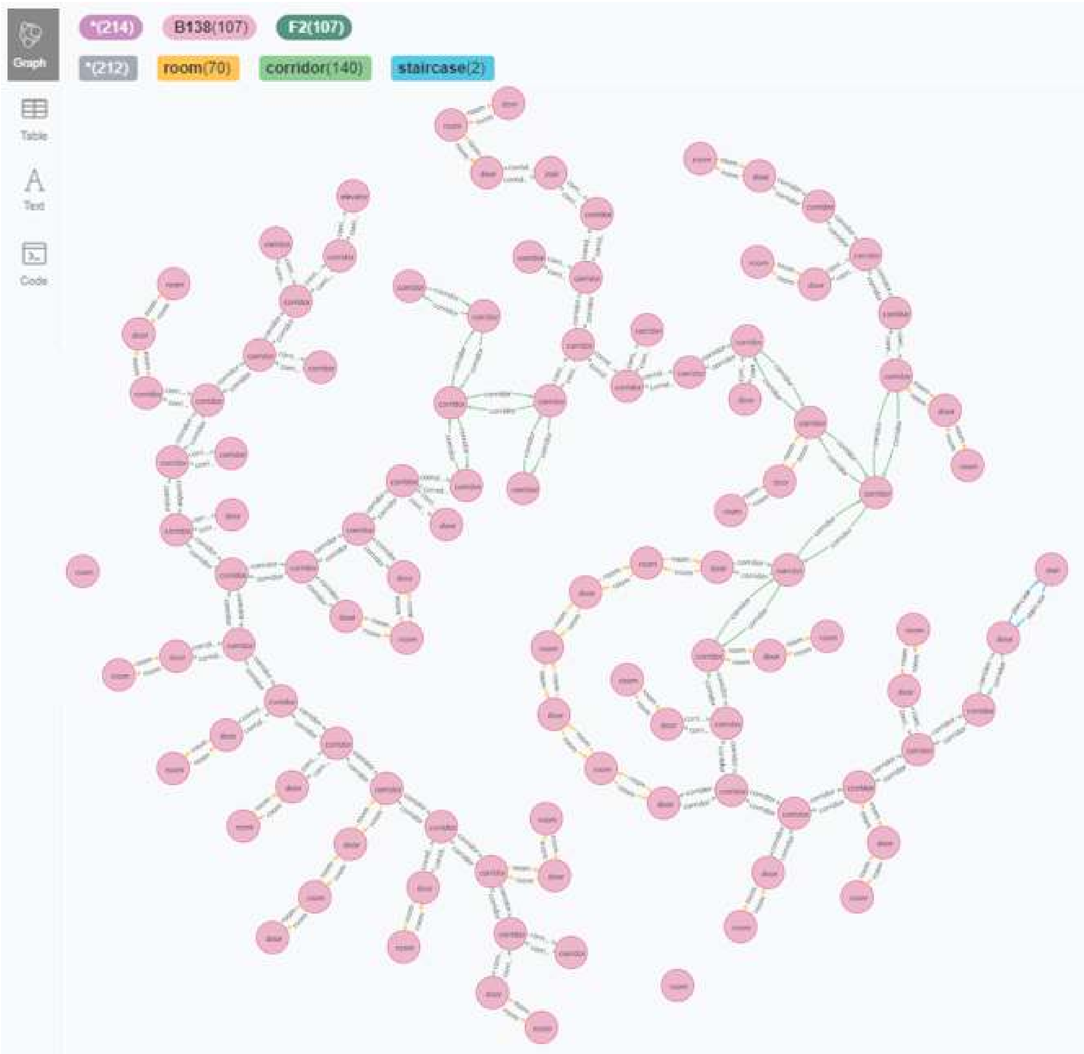
[Figure B-2] The generated indoor graph database for set-B



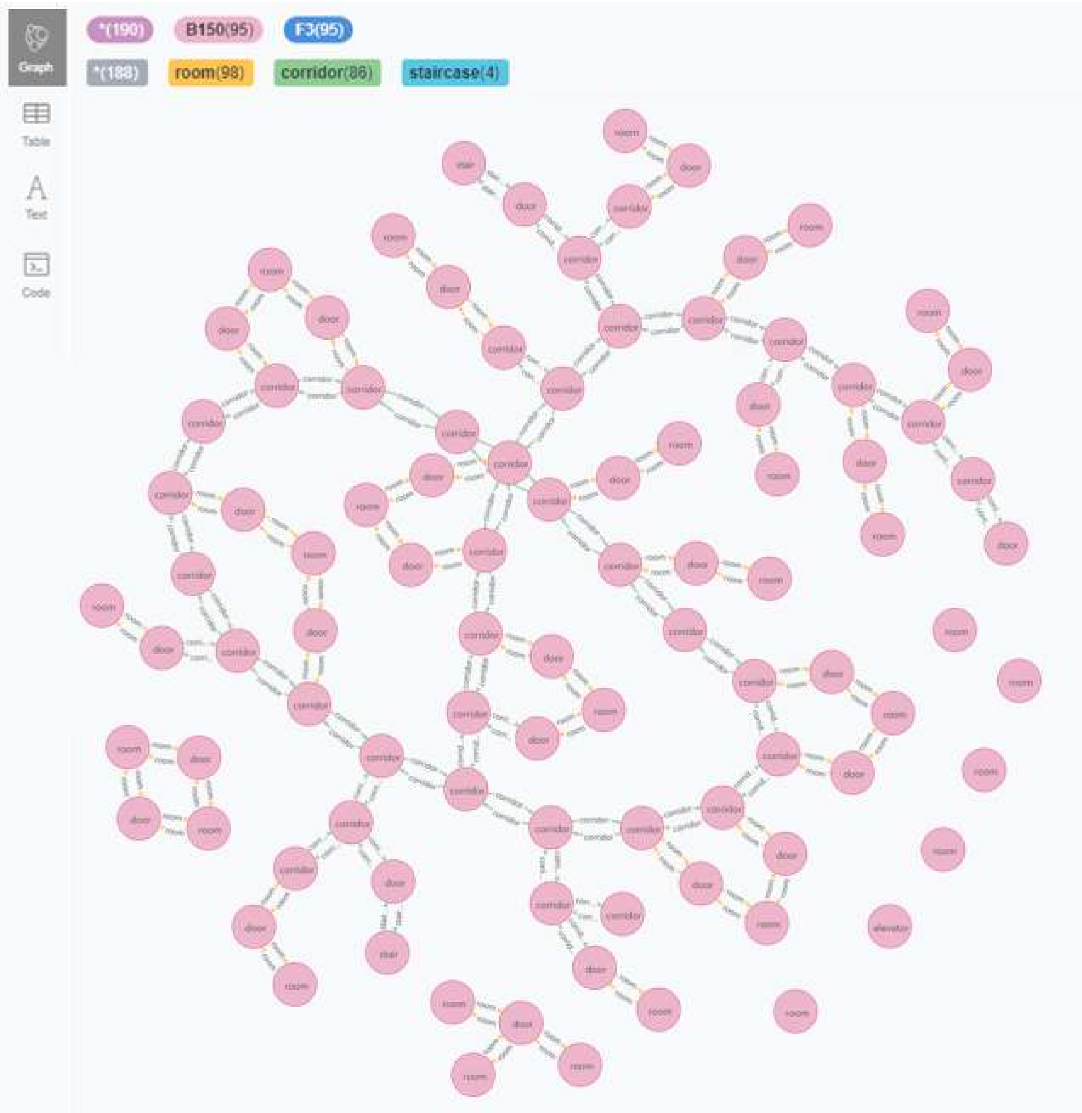
[Figure B-3] The generated indoor graph database for set-C



[Figure B-4] The generated indoor graph database for set-D



[Figure B-5] The generated indoor graph database for set-E



[Figure B-6] The generated indoor graph database for set-F

국 문 초 록

사람들의 실내 활동이 다양해지면서 건물의 규모가 커지고 구조가 복잡해지고 있다. 이러한 실내 환경의 변화는 교통약자의 이동성 보장에 대한 사회적 관심을 증가시켰으며, 교통약자 맞춤형 실내 라우팅 서비스에 대한 수요 또한 증가시켰다. 특히 많은 이동 제약을 가지는 이동약자 대상 서비스의 경우에는, 최적 경로를 계획하는 과정에서 개인의 선호나 경험이 반영된 개인화된 서비스로 범위가 확장되고 있다. 이러한 배경에서, 스키마가 유연하고 데이터의 가공 및 처리가 효율적인 데이터베이스의 구축이 필요하다.

본 연구에서는 스캔한 도면 이미지를 활용한 이동약자용 실내 그래프데이터베이스 구축 기법을 제안하였다. 먼저, 국내외 실내 공간 관련 표준 및 설계 기준들의 검토를 통해 이동약자의 통행과 관련된 실내 공간 및 객체, 영향 요인들을 도출하여 개념적 데이터 모델을 설계하였다. 또한, 실내의 각 공간과 시설물의 기하정보와 위상정보를 기반으로 이동약자의 접근성 및 통행 가능성을 정량화하기 위한 접근성 지수를 설계하였다. 다음으로, 스캔 도면을 입력하여 이동약자용 실내 그래프 데이터베이스 구축을 위한 프로세스를 제안하였다. 제안한 프로세스는 전이학습 기반 접근 방식을 통해 스캔 도면에서 공간의 구조 정보를 추출하고, 토폴로지 추출 및 접근성 평가를 통해 이동약자용 네트워크 모델을 생성하며, 생성한 네트워크 모델을 그래프 데이터베이스로 자동 변환하는 과정을 포함한다. 구체적으로, 제안 프로세스는 수정된 ResNet 기반의 모델을 새롭게 라벨링한 도면으로 미세 조정하여 사용함으로써 실내 구조맵을 생성한다. 이후 추출된 객체들의 공간 관계를 기반으로 각 공간을 노드와 링크로 표현한 실내 네트워크 모델을 구축한다. 각 공간의 접근성 정보는 제안된 접근성 지수와 임계값을 사용하여 생성된 후 데이터베이스에 저장되어, 이동약자를 위한 접근 가능한 그래프 추출에 활용될 수 있다.

본 연구에서는 제안한 기법을 서울대학교 도면 데이터 셋에 적용하여 이동약자용 실내 그래프 데이터베이스를 구축하고 평가하였다. 구축한 실내 그래프 데이터베이스를 활용하여 다층 경로 계획과 실내외 연계 경로 계획의 2가

지 시나리오에 따라 최적 경로를 도출하였다. 그 결과, 일반 보행자의 최적 경로와 비교하여 이동약자용 최적 경로는 가까운 계단이 아닌 엘리베이터를 통한 수직 이동을 포함하였을 뿐만 아니라 접근 불가능한 공간을 회피하도록 도출되었다. 즉, 제안한 기법을 통해 이동약자 측면에서 통행 장애 정보를 포함하여 실내 환경을 적절하게 묘사하는 데이터베이스의 구축이 가능함을 확인할 수 있었다. 또한, 출입로로 명명된 관계 생성만으로 스케일이나 좌표 변환 없이 실내외 연계 경로 계획이 가능하였는데, 이는 독립적인 데이터 간 연계 사용에 적합한 그래프 데이터베이스의 특성을 반영한 결과로 판단할 수 있다.

본 연구의 주요 기여는 스캔한 도면을 사용하여 이동약자용 실내 그래프 데이터베이스를 구축하기 위한 프로세스를 개발한 것이다. 구체적으로, 이동약자의 이동에 초점을 두고 설계한 데이터 모델을 기반으로 한 데이터베이스 구축이 가능하므로 이동약자용 실내 길안내 서비스에 활용될 수 있다. 또한, 토폴로지 구축 및 그래프 데이터베이스로의 변환을 위한 하위 프로시저를 개발하였으며, 제안 프로세스는 해당 프로시저들로 구성되어 도면 입력을 통해 이동약자용 실내 그래프 데이터베이스 구축을 가능하게 한다. 해당 하위 프로시저들은 자동으로 수행될 수 있어 데이터베이스 구축 시 소요되는 시간과 비용을 절감할 수 있다. 또한, 다양한 정형 및 비정형 데이터의 연계에 적합한 그래프 데이터베이스의 특징에 의해, 제안한 프로세스를 통해 구축한 실내 데이터베이스는 기존 공간 모델의 기능을 포함하면서 다양한 유형의 길안내 서비스에 활용될 수 있을 것으로 기대된다.

주요어 : 그래프 데이터베이스, 전이학습, 네트워크 모델, 데이터 모델, 이동약자, 접근성

학 번 : 2017-31313