Hindawi
Advances in Civil Engineering
Volume 2024, Article ID 8839865, 17 pages
https://doi.org/10.1155/2024/8839865

Research Article

A Semantic Approach to Dynamic Path Planning for Fire Evacuation through BIM and IoT Data Integration

Bo Pang[1,2] **Jianyong Shi**[1,2] **Liu Jiang**[1,2,3] **and Zeyu Pan**[1,2]

[1]*School of Naval Architecture, Ocean and Civil Engineering, Shanghai Jiao Tong University, Shanghai, China*
[2]*Shanghai Key Laboratory for Digital Maintenance of Buildings and Infrastructure, Shanghai, China*
[3]*School of Civil Engineering and Architecture, Hainan University, Haikou, China*

Correspondence should be addressed to Jianyong Shi; shijy@sjtu.edu.cn

Received 1 November 2023; Revised 1 March 2024; Accepted 12 March 2024; Published 17 April 2024

Academic Editor: Tiago Ferreira

Fire evacuation path planning involves multiple data sources. In order to develop a dynamic planning, a comprehensive knowledge of the environment involving building information and fire development is required. This article presents a semantic approach that integrates Building Information Modeling (BIM) and Internet of Things (IoT) information to provide a data foundation for dynamic path planning. First, a fire evacuation (FE) ontology is introduced to fuse both knowledge and information relevant to dynamic path planning. Next, a dynamic knowledge graph that evolves according to the development of fire situation is instantiated based on the relevant FE ontology. Finally, to validate the feasibility of the semantic approach based on the ontology and knowledge graph, an example of application is conducted using a specific building as an example. This study provides a data foundation for more intelligent and precise decision-making in fire evacuation scenarios and offers a new approach for safety design and management in the field of construction.

1. Introduction

As urbanization continues to advance, the complexity of interior building structures and population density have increased. The enrichment of building functions has led to an increase in the time people spend indoors. According to a report from the United States Environmental Protection Agency (EPA), humans spend approximately 90% of their time indoors [1]. Meanwhile, fires are one of the most common and severe emergencies that occur indoors [2]. The Grenfell Tower Fire occurred in London in 2017 stands as a profound tragedy, which engulfed the entire building, resulting in the loss of 80 lives and injuring over 70 individuals [3, 4]. The National Fire Protection Association (NFPA) reports an increase in casualties and property losses due to fires in the past decade. In 2022 alone, there were over 1.5 million fire incidents in the United States, resulting in 3,790 fatalities, 13,250 injuries, and $18 billion in property damage [5]. Additionally, the National Fire and Rescue Administration of China reported that the fire in 2022 resulted in 1,381 deaths, 2,063 injuries, and approximately $1 billion in direct economic losses [6]. The above data underscore the importance of conducting research on fire emergency (FE) evacuation.

When a fire occurs, it is crucial for affected individuals to evacuate quickly [7]. Insufficient knowledge of the building's structure can impact the efficiency of evacuations [8]. Proper evacuation paths can guide individuals to safety in the shortest time possible, thereby reducing casualties [9]. In traditional fire evacuations, the location of the fire source is determined based on human reactions [10]; additionally, according to the International Standard ISO 23601 [11], the evacuation paths marked on evacuation plans within buildings generally guide individuals to evacuate from the nearest exits, and adjustments to evacuation paths based on the development of a fire are not permitted. The above process has issues such as long response times, low evacuation efficiency, and delayed responses to changing fire developments, often resulting in more significant casualties [12]. Therefore, gaining a comprehensive understanding of the entire building space and the real-time fire development, as defined in the International Standard ISO 20414 [13] as "environment," is crucial for enhancing evacuation efficiency and ensuring the safety of individuals during evacuation.

In comparison to traditional manual operations, the emergence of the Internet of Things (IoT) provides a means to sense the environment and collect data through intelligent device networks. It enables the early detection of fires [14] and the real-time collection of sensor data to monitor fire development [15]. Building Information Modeling (BIM), an emerging technology emphasizing a 3D model containing all building information [16], can provide comprehensive information for evacuation path planning. The integration of BIM and IoT can present real-time environmental information and spatial status during fire scenarios [17], supporting evacuation path planning and real-time updates based on fire development. For instance, Chen et al. [18] developed a fire situation awareness framework based on BIM models and IoT devices. It rapidly locates the fire source using IoT data and creates virtual fire environments using augmented reality (AR) and virtual reality (VR) for training the situational awareness of rescue personnel. Li et al. [19] proposed a BIM-based rapid person localization and graphical interaction method that utilizes IoT devices to sense the environment and provide geometric information as an input for spatial segmentation algorithms. They evaluated the algorithm's accuracy using simulation methods. Chen et al. [20] proposed a fire rescue visualization and warning system based on BIM and FDS, utilizing IoT technology to monitor the fire situation in real time and provide evacuation route guidance based on the fire's development. However, the above-mentioned studies lack integration and interaction means for handling multisource data, such as BIM, IoT, and personnel information. Data collected from different fields are often dispersed and independent, with weak interconnections and lacking synchronization, posing difficulties and obstacles for further intelligent information and data processing.

Ontologies formalize and structure the description of shared knowledge [21]. They help establish formal definitions of different concepts in the field of fire evacuation and clarify their relationships. As a result, ontologies assist humans and computers in understanding and utilizing knowledge in the field of fire evacuation [22]. The communication and interoperability between multisource information from BIM, IoT, and other sources can be improved by creating ontologies for shared understanding [23]. Knowledge graphs built based on ontologies store and retrieve knowledge about entities, relationships, and attributes in the form of triplets. The SPARQL protocol and RDF query language [24] can be implemented to achieve efficient retrieval of building information [25]. Li et al. [26] created an emergency response ontology, representing the semantics of FE response workflows, allowing information systems to infer the subsequent steps to be taken. The development of the SEMA4A ontology [27] was initially for conveying emergency notifications using various technologies in different emergency situations, later expanding to concepts related to the accessibility of evacuation routes for people in emergency situations. Nuo et al.'s [28] ontology was used to generate semantic maps of buildings in FE situations, enabling the acquisition of information on smoke propagation and escape routes to support rescue personnel and victims during the rescue process. However, the goals of these ontologies are relatively general, and they lack detailed descriptions of core information in the fire evacuation field. There is also limited research on data extraction methods for building environmental information, making their application in fire evacuation path planning scenarios challenging.

To address the issues in information integration and interaction in the field of fire evacuation, this study proposes a comprehensive semantic approach to achieve precise data extraction and interoperability from multiple sources, thus supporting more intelligent fire evacuation path planning in fire scenarios. This study presents a dedicated FE ontology for the field of fire evacuation that integrates BIM and IoT information, and constructs a knowledge graph based on this ontology. A multistory building's fire scenario is used for path planning as an example to validate the practicality and feasibility of the proposed ontology and knowledge graph.

2. Background and Framework

The dynamic planning process for evacuation routes in a fire scenario involves multiple sources of data, including the number of people, the development of the fire, and architectural spatial information. This section will introduce the data involved and propose a semantic approach to integrating BIM and IoT data.

2.1. Data in Path Planning. In the field of fire safety, applicable path planning algorithms can generally be divided into two categories [29]: (1) traditional algorithms, represented by the Dijkstra [30] algorithm and the A^* [31] algorithm, and (2) intelligent algorithms such as the heuristic algorithm [32] and reinforcement learning [33]. Traditional algorithms typically rely on known environments and fixed rules for path planning. For example, the Dijkstra algorithm is a widely used single-source shortest path algorithm in directed weighted graphs, while the A^* algorithm is a heuristic search algorithm that combines the characteristics of best-first search and Dijkstra algorithm, suitable for solving path planning problems with heuristic information. Intelligent algorithms, on the other hand, are more flexible and adaptable to uncertain environments and dynamic changes. Heuristic algorithms, represented by Ant Colony Algorithm (ACO) [34], use heuristic functions to guide the search process to efficiently find the target. Reinforcement learning algorithms learn the optimal behavior policy through interaction with the environment, making them suitable for path planning scenarios that require real-time learning and adaptation to the environment.

Despite differences in the functions and iteration methods used by various algorithms, they all require scene information, including the connectivity between rooms as optional paths and the distance from the current location to the target location as a basis for selection. Taking the Dijkstra algorithm as an example, it is typically used to find the weighted shortest path from the starting point to

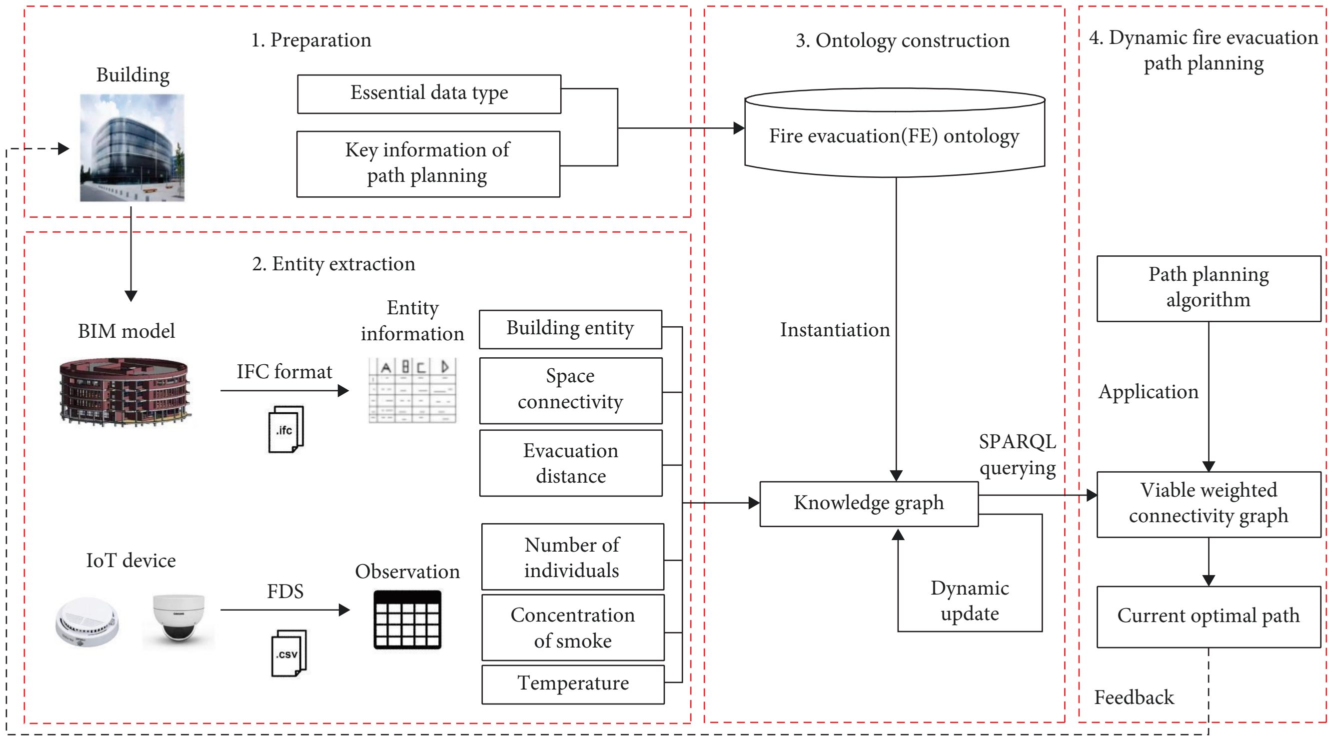


Figure 1: Theoretical framework for the semantic approach proposed.

the destination in a weighted network graph [30]. In the application of the Dijkstra algorithm for path planning in the context of fire safety, the first step is to obtain a weighted topological connectivity graph of the building space. This involves all spatial information within the building, their interconnections, and the evacuation distances between connected spaces. In this study, the aforementioned information are integrated using FE ontology and a weighted topological graph is constructed to provide data foundation for path planning algorithms.

2.2. Data in Fire Observation. Due to the expansibility and volatility [2] of fires, and their rapid development within complex indoor environments [12], conducting path planning only at the beginning of an evacuation may not adequately meet the evacuation requirements in a fire scenario. Therefore, continuous monitoring of the evolving fire situation is necessary, and path planning should be revised as needed to ensure that evacuation routes remain unaffected by the fire [35], thereby protecting evacuees from harm. To achieve this, it is essential to use IoT devices to observe relevant information [15] related to the fire's development, including the number of people awaiting evacuation, visibility, temperature, CO concentration, and more. Among these, the number of people awaiting evacuation in a specific space is used to determine the starting point for path planning, visibility is used to determine the time of fire alarm, and temperature and CO concentration are used to assess whether the observed space is affected by the fire and should not be used for evacuation. Therefore, the FE ontology in this study is built to integrate the mentioned data, and the knowledge graph based on FE ontology is updated accordingly as the fire evolves, thereby providing dynamic evacuation path planning.

2.3. Theoretical Framework. The theoretical framework of this study is depicted in Figure 1. It consists of four main stages. First, in the preparation stage, a BIM model of the actual building is established, and essential data type related to fire evacuation and key information required for path planning are classified and organized for the subsequent construction of the fire evacuation ontology. Second, in the entity extraction stage, entity information is extracted from the IFC standard format files of the building information model, covering building entities, spatial relationships, evacuation distances, etc. Additionally, FDS are applied to simulate the observation of IoT devices in real fire scenarios, focusing on results such as the number of individuals, concentration of smoke, temperature, etc. Subsequently, in the ontology construction stage, the fire evacuation ontology is built by combining data types and information mentioned in Sections 2.1 and 2.2. Entity information and observation results are instantiated into a knowledge graph format. When the fire development situation changes, the content of the knowledge graph is dynamically updated. Finally, using the SPARQL language, information required for path planning is queried to construct a weighted connectivity graph. Combined with specific path planning algorithms, the optimal path at the current moment is calculated, and the results are used to guide the evacuation behavior of personnel in actual buildings. Through this framework, the process of dynamic path planning for fire evacuation scenarios becomes more systematic and automated, providing strong support for improving evacuation efficiency and ensuring personnel safety.

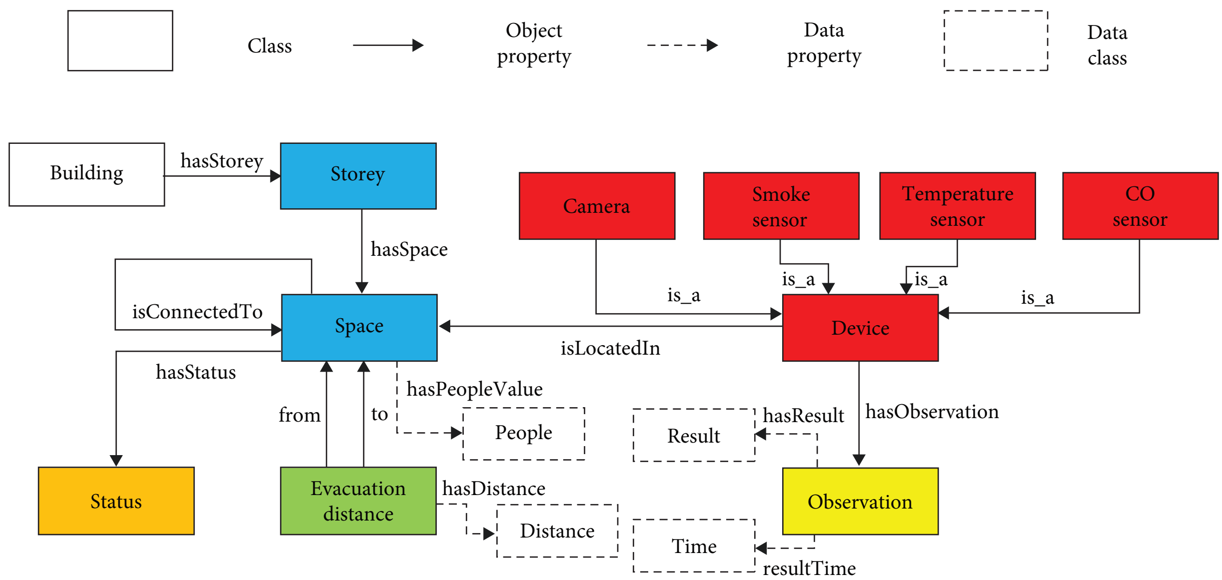


FIGURE 2: Structure of FE ontology.

The remaining parts of this paper are organized as follows: Section 3 introduces the structure of the FE ontology. Section 4 provides detailed information on the construction of knowledge graph involving data extraction and dynamic update. Section 5 showcases the results and evaluations in specific scenarios, and the final section draws conclusions and suggests future research directions.

3. Structure of FE Ontology

The fundamental building blocks of an ontology include classes and properties [21]. In an ontology, classes are used to categorize and organize entities within a domain. The hierarchy of classes can form a classification system that defines hierarchical relationships between entities [36]. Properties are divided into object properties and data properties [37]. Object properties are employed to express logical relationships between entities of different classes, while data properties are used to define attributes and associated parameters, offering descriptive information about entities.

The structure of FE ontology established in this study is depicted in Figure 2. Within the context of fire evacuation, space and devices are central to dynamic path planning, as highlighted in Figure 2. Spaces provide architectural spatial information necessary for path planning, while devices offer real-time observational data and the data are updated according to the development of fire in order to support the dynamic path planning as fire evolves. The entire process of ontology construction was carried out using the Protégé.

To begin with, a hierarchical structure, "Building—Storey—Space", is defined to determine the specific storey on which a particular space is located within a building. The "hasStorey" property specifies the storeys a building has, while the "hasSpace" property determines space situated in a certain storey. Additionally, spatial connectivity is defined by the "isConnectedTo" property, and evacuation distances between two space entities are represented by a pair of object properties, "from" and "to," along with the "EvacuationDistance" class. The "Status" class is an enumerated class encompassing instances "Normal" and "Dangerous" to determine whether a space is in safety status. Furthermore, data properties are defined including "hasPeopleValue" for "Space" to indicate the number of unevacuated individuals in a space and "hasDistance" for "EvacuationDistance" to express the evacuation distance between two space entities.

Regarding devices, four subclasses are defined under the "Device" class, representing the four common types of IoT devices involved in the fire evacuation process. The property "isLocatedIn" specifies the space where a device entity is located. The description of the observation of devices is inspired by the sensors, observations, samples, and actuators ontology [38]. The "hasObservation" property defines the observational result of a device entity, while data properties "hasResult" and "resultTime", respectively, specify the observation time and value for entities of the "Observation" class.

4. Construction of Knowledge Graph for Evacuation

A knowledge graph can be regarded as a knowledge database based on an instantiation of the relevant ontology, showcasing entities and their relationships in a graphical form [39, 40]. In this section, entity extraction methods relevant to fire evacuation are introduced.

4.1. Extraction of Building Information. In practical applications, BIM models can be created using various modeling software [41], which may result in multiple data formats. Industry foundation classes (IFC) is an open standard for data exchange in the BIM domain [42]. BIM model information created by different types of software can be standardized and stored in the IFC format using the EXPRESS language [43]. Building information can be extracted from IFC files, and then mapped to the FE ontology based on the

Table 1: Correspondence between the IFC schema and the FE ontology.

IFC schema	FE ontology
IfcBuilding	“Building”
IfcBuildingStorey	“Storey”
IfcSpace	“Space”
IfcAlarm	“Smoke Sensor”
IfcBuildingElementProxy	“Camera”

IFC schema. Furthermore, linked data can be defined in the knowledge graph based on the relationships between IFC entities. In this study, IFC4 data format is used and IfcOpenShell and PythonOCC libraries are employed for file parsing and geometric operations.

4.1.1. Extraction of Building Entities. Initially, entities corresponding to “Building”, “Storey”, “Space”, and “Device” are extracted by parsing the IFC file. The mapping relationship between the IFC schema and the classes in the ontology is presented in Table 1. The by_type function from the IfcOpenShell library is used to extract building entities based on their types. Subsequently, these entities are populated into the corresponding classes in the ontology according to the mapping relationship. The entities are named consistently with the identifiers in the IFC file. For instance, an entity named IfcAlarm_954929 in IFC will be retained with the suffix identifier and named as Alarm_954929 in the knowledge graph.

4.1.2. Extraction of Space Connectivity. In IFC4, a space is defined as an instance of the IfcSpace class, representing a region or volume with theoretical or practical boundaries [44]. The boundaries of a space in IFC4 are defined by the IfcRelSpaceBoundary, which is considered as a link between elements and spaces, realizing the description and management of relationships between elements and spaces [44]. Elements can be either virtual or physical, and two IfcRelSpaceBoundary instances connected by a virtual element indicate that two IfcSpace entities are virtually connected. Similarly, two IfcRelSpaceBoundary instances connected by physical elements signify that two IfcSpace entities are adjacent and separated by wall entities, such as IfcWall or IfcWallStandardCase. When the same physical space is divided into multiple parts in the BIM model, and each part is defined as an instance of the IfcSpace class, these entities are considered having theoretical boundaries and being virtually connected, for example, different parts of the same stairwell on different storeys. However, for physically bounded areas, during the extraction of space connectivity, it is necessary not only to assess their spatial adjacency, but also to determine if there are doors connecting the spaces through elements. This is crucial for selecting evacuation paths during an emergency.

The logic for extracting space connectivity is shown in Figure 3. In Figure 3(a), the criterion for determining virtual connectivity between Space_1 and Space_2 is established through the boundaries of Space_1 and Space_2 connected by an instance of IfcVirtualElement. In Figure 3(b), Space_1 and Space_2 are connected by a physical element of IfcWall class, and the logical chain “Wall-> Opening-> Opening Element-> Filling Element-> Door” determines that Space_1 and Space_2 are connected by a door. The entire process is implemented using Python code based on IfcOpenShell, and the extracted connectivity relationships are written into the knowledge graph in the form of triples: <“Space entity”, isConnectedTo, “Space entity”>.

4.1.3. Extraction of Evacuation Distances. Evacuation distance is a critical parameter in fire evacuation scenarios. When conducting path planning, it is essential to select the shortest route for evacuation. According to the “Architectural Design Fire Code”, evacuation distance is calculated considering the distance from the least favorable point inside a space to the exit, while accounting for obstructions of walls [45]. The actual calculated distance is represented as a polyline distance. In this study, the evacuation distance is computed using the following method:

(1) For spaces physically connected as mentioned in Section 4.1.2, the boundaries of spaces and the positions of doors are obtained using PythonOCC library. The gt_Pnt function is used to retrieve vertices.
(2) Distances between vertices within the space and distances from vertices to doors are calculated, noted as L_P and L_D, respectively. First, connect all vertices and doors in the space. The length of the connection represents the straight-line distance between the two points; when the connection line exceeds the boundary of the space, delete the connection line, as shown in Figure 4. The distance between vertices are noted as L_{Pij}, where i, j represents the mark of vertices, and the distances between a vertex and the door are noted as L_{Di} similarly.
(3) The distance from the least favorable point to the door is determined and considered as the evacuation distance from the space to an adjacent space. For vertices that do not have a direct connection line to the door, the shortest polyline distance is considered, i.e., $L_{Ci} = \min(L_{Pij} + L_{Di})$. The final evacuation distance L is obtained as $L = \max(L_{Ci}, L_{Di})$ and the result is written into the knowledge graph in the form of triples: <“Evacuation Distance Entity”,from, “Space Entity 1”>, <“Evacuation Distance Entity”, to, “Space Entity 2”>, <“Evacuation Distance Entity”, hasDistance, evacuation distance L>.

4.1.4. Extraction of Device Locations. Information regarding the location of devices, specifically the entities within the “Device” class in FE ontology, can be determined by intersecting the geometric shapes of devices and spaces using the PythonOCC library. Initially, the by_type function from the IfcOpenShell library is used to select space entities and device entities associated with the building storey entity IfcBuildingStorey, as depicted in the logical relationship shown in Figure 5. Subsequently, geometric processing is performed

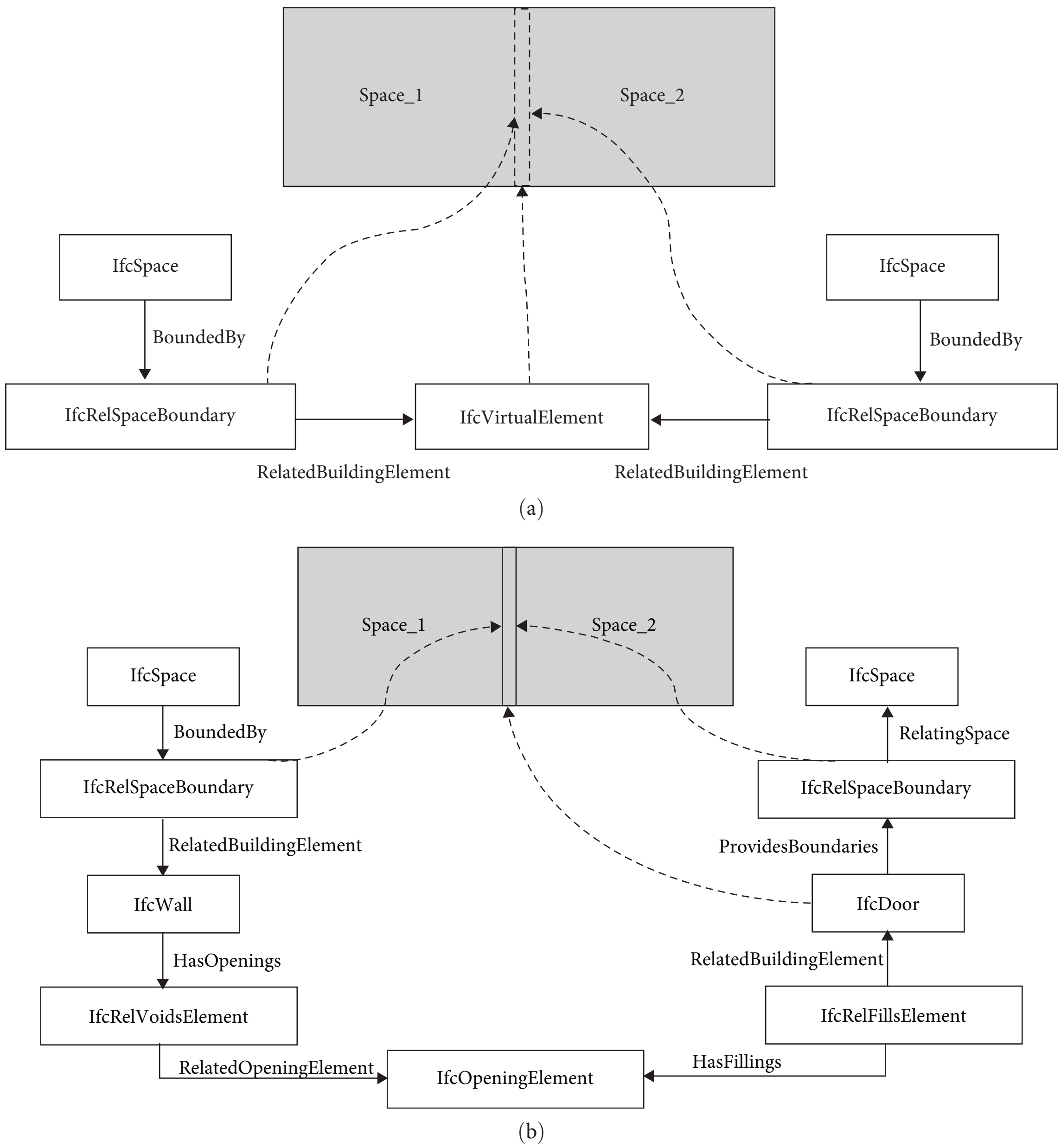


FIGURE 3: Obtaining spatial relationships of virtual connectivity (a) and doorway connectivity (b).

using the PythonOCC library to obtain the topological shapes of space entities and device entities. These entities are then translated along the *Z*-axis to the same plane. The BRepAlgoAPI_Common function is employed to conduct Boolean intersection operations on the topological shapes. If the intersection is nonempty, then it confirms that the device entity is located within the space entity. Taking Figure 6 as an example, the space entity IfcSpace_3205 and the device entity Alarm_855609 have a nonempty intersection, allowing us to represent this relationship in the knowledge graph in the form of a triple: <Alarm_855609, isLocatedIn, IfcSpace_3205>. This indicates that the device entity is located within the space entity IfcSpace_3205. Conversely, the device entity Alarm_855677 is not located within this space entity.

4.2. Processing of IoT Devices Observation. In the field of fire evacuation, data obtained from IoT devices observation plays a crucial role in dynamic path planning [17]. In this section, the data encompass two main aspects: image-based observations and fire-related observations. By applying image recognition algorithms to analyze the number of individuals in scene images, the number of evacuees in the space where the corresponding camera device is located can be determined. On the other hand, fire-related data, such as temperature, visibility, and CO concentration, can be used to pinpoint the time of fire alarms and assess the safety of the space where the IoT devices are placed based on the characteristics of the fire development stages. The integration and analysis of these observational data contribute to ensuring the efficient execution of fire evacuation plans, thereby safeguarding lives and property.

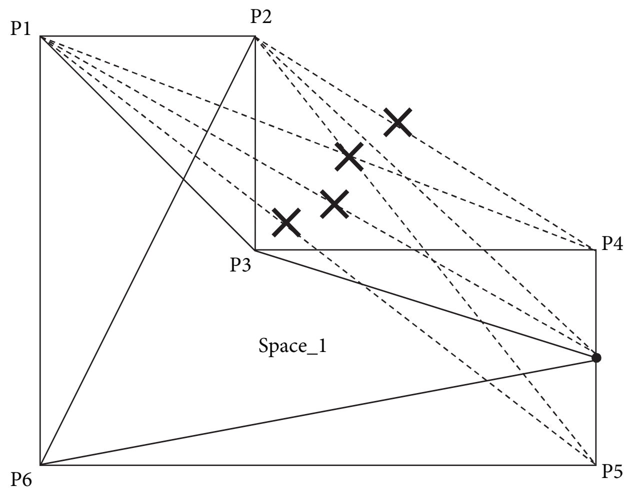


FIGURE 4: Example of calculating evacuation distance.

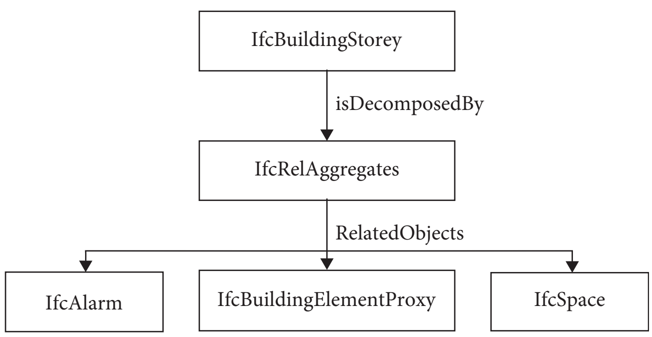


FIGURE 5: Extracting devices and spaces located in a specific storey.

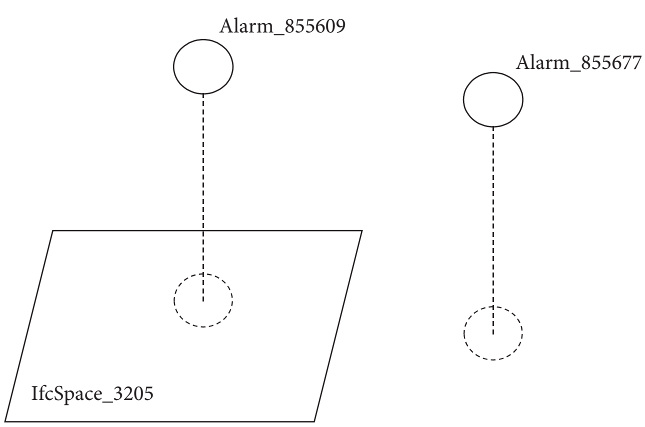


FIGURE 6: Example of calculating the intersection between space and device.

4.2.1. Processing of Camera Observation. To determine the number of individuals awaiting evacuation within a specific space, scene images captured by cameras can be processed using image recognition algorithms. Vora and Chilaka [46] proposed a method for fast and accurate human head detection, known as Fast and Accurate Head Detector (FCHD). The convolutional neural network (CNN) architecture for FCHD is shown in Figure 7. In this method, standard format input images first undergo preliminary feature extraction using a pretrained VGG16 network. Subsequently, these features are encoded through a convolutional layer and then pass through separate convolutional layers with 1×1 convolution kernels to enter the regression head and classification head. This dual strategy aims to simultaneously predict the probability score of human heads and their precise spatial coordinates. Postprocessing techniques include bounding box transformation and nonmaximum suppression to obtain the spatial coordinates of detected human heads. The number of convolution kernels used by the classification head and regression head is determined by the predetermined number of anchors (bounding boxes), denoted as N. The neural network is trained and validated on the BRAINWASH public dataset.

To make the image recognition algorithm more suitable for fire scenarios, this study performs data augmentation on the training dataset. By introducing a foggy effect, it simulates images captured in real fire scenes. These enhancements are aimed at improving the accuracy of human head detection in fire environments. The foggy effect can be achieved by manipulating the RGB channels. The addweighted function from the OpenCV library is used to blend the original image with a custom grayscale image. By adjusting the weight of the grayscale image, the intensity of the foggy effect can be controlled. A comparison of the training dataset images before and after the enhancement is shown in Figure 8.

The improved dataset is used to retrain the FCHD network. The initial parameters for VGG16 are obtained from the Imagenet pretrained model, while subsequent network parameters were initialized using a standard normal distribution with a variance of 0.01. The loss function used in this study is the same as in [22], and parameter adjustments are performed using stochastic gradient descent with an initial learning rate of 0.001. A total of 15 training epochs are conducted, and after the initial eight epochs, the learning rate is reduced by a fixed rate of 0.1. The model's weight is saved at the end of each training epoch. Model performance is evaluated based on the average precision (AP) on the test set, where AP is defined as the ratio of the number of bounding boxes with an intersection over union (IoU) greater than or equal to 0.7 after nonmaximum suppression to the number of annotated human heads.

Figure 9 shows the changes in the regression loss function (Figure 9(a)) and classification loss function (Figure 9(b)) during the training process. The loss functions exhibit some instability in the initial stages of each training epoch, but after 15 epochs, the model gradually stabilizes. The average prediction accuracy on the test set at the end of each training epoch is presented in Table 2. The model with the highest average prediction accuracy, obtained in the 9th epoch (Epoch 9), achieves an average accuracy of 0.88 on the test set, enabling accurate estimation of the number of people in the room. The recognition results are represented in the knowledge graph as triples: <"Camera Entity", hasObservation, "Observation Entity">, <"Observation Entity," hasResult, number of people >. Additionally, the observation time is recorded, and a

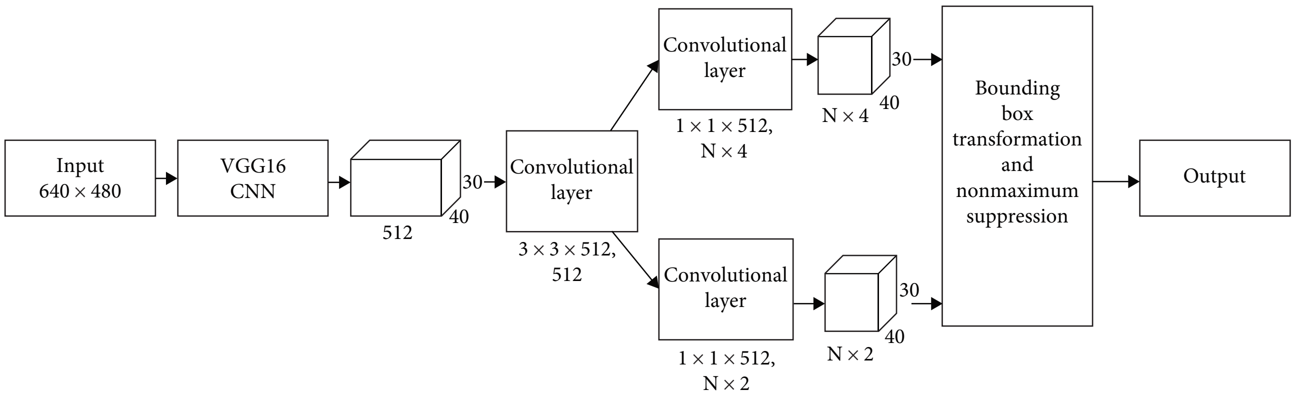
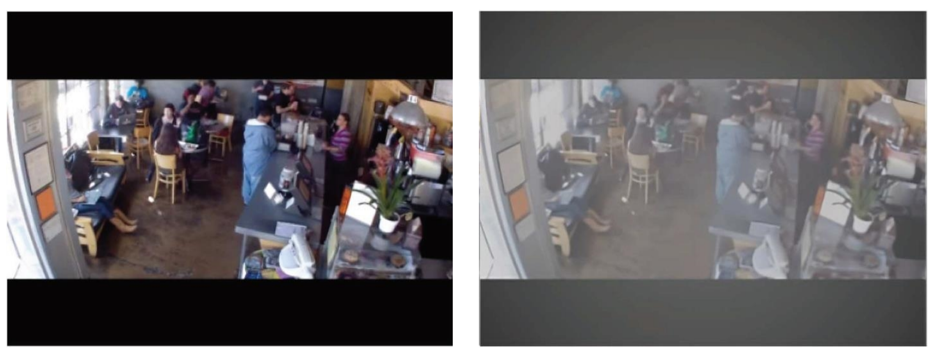
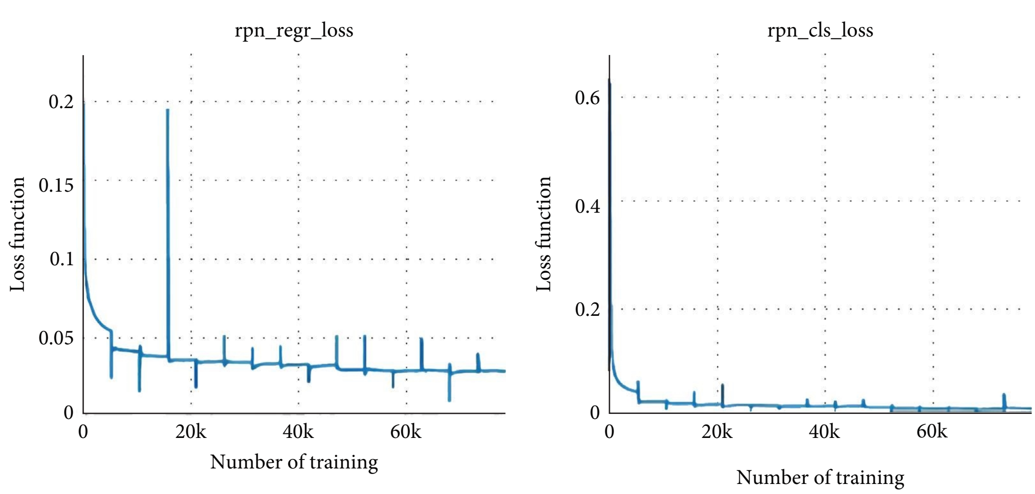


Figure 7: Structure of FCHD neural network.



(a) (b)

Figure 8: Comparison of data set before (a) and after (b) process.



(a) (b)

Figure 9: Regression loss (a) and classification loss (b).

data attribute <“Space Entity,” hasPeopleValue, number of people > is added to the space entity.

4.2.2. Fire-Related Observation. Smoke sensors, temperature sensors, and CO sensors distributed in the building can observe visibility, temperature, and CO concentration in fire scenarios, respectively. Smoke sensors are used to trigger fire alarms, alerting individuals in the environment to evacuate when the light obscuration exceeds 0.0328/m. Temperature sensors and CO sensors assess room safety and passability by observing fire-related data. The observation results are stored in the knowledge graph in the form of

Table 2: Accuracy in the training process of neural networks.

Epoch	Validate average accuracy
1	0.680
2	0.836
3	0.863
4	0.753
5	0.872
6	0.739
7	0.872
8	0.879
9	0.880
10	0.762
11	0.754
12	0.760
13	0.765
14	0.770
15	0.782

Table 3: Fire hazard rating table.

Hazard level	Temperature (°C)	CO concentration (ppm)
Safe	<42	<50
Potentially dangerous	42–50	50–200
Dangerous	50–80	200–2,000
Lethal	>80	>2,000

Figure 10: BIM model of building example.

triples: <“Device Entity,” hasObservation, “Observation Entity”>, <“Observation Entity,” hasResult, observation value >, along with the recording of observation time: <“Observation Entity,” resultTime, observation time >. Based on these data, the time of fire alarm triggering can be determined, as well as the threat level of the device’s space to the evacuating crowd at a given moment.

4.3. Dynamic Update Mechanism of Knowlegde Graph for Evacuation. In order to adapt to the dynamic evolution of a fire incident, this study dynamically updates the knowledge graph based on IoT observation data. It primarily focuses on the data results “result” and timestamps “time” of the “Observation” entities to assess the safety status of “Space” entities. If an IoT device detects a hazardous state in a specific “Space” entity, the associated “Status” entity is updated to “Dangerous”, and the data property of “Evacuation Distance” entity associated is set to a significantly large value, thereby preventing the path planning algorithm from routing through this hazardous space.

The progression of a fire generally involves three stages: initial growth, full development, and decay [47]. Based on the patterns of temperature and CO gas concentration during these stages, and considering the maximum duration a person can tolerate in different temperature and CO concentration environments, the fire scenario is classified into four hazard levels [48], as shown in Table 3. According to the hazard level table, when the environment temperature is between 50°C and 80°C or CO gas concentration is between 200 and 2,000 ppm, individuals who have not evacuated are likely in a dangerous state. Therefore, in this study, spaces with temperature observation values above 60°C or CO gas concentration observation values above 1,000 ppm are designated as impassable areas. In the knowledge graph, the state entity for IoT devices located in spaces with observed hazardous values is updated to “Dangerous.” The evacuation distance value from adjacent space entities to this space is updated to 1,000 m, indicating that this space should be avoided in the path planning algorithm. By applying the above rules to inference based on the knowledge graph, relevant observation values can be selected from the knowledge graph.

5. Results and Discussion

In this section, a specific multistory building is selected to validate the feasibility of the semantic approach for dynamic fire evacuation path planning proposed in this study, and the results are presented and discussed. First, in the preparation stage, the BIM model established and the parameter settings of FDS are introduced. Subsequently, based on the FE ontology established in Section 3 and the methods outlined in Section 4, a dynamic knowledge graph is established to provide prior knowledge of fire evacuation. Considering the evolving fire scenario, the information stored in the knowledge graph is automatically updated. Next, a set of SPARQL queries is designed to extract the information required for path planning from the knowledge graph to construct a topological connectivity graph. It is worth noting that, due to the dynamic nature of the knowledge graph, the constructed connectivity graph also dynamically changes with the development of fire. Evacuation path planning is then conducted, and compared with evacuation behaviors without the FE ontology and knowledge graph to validate the advantages of the semantic approach proposed in this study.

5.1. Preparation. In this study, a four-storey library is selected as object. Autodesk Revit 2022 is used to establish the BIM model of the library, as shown in Figure 10, and is exported as an IFC file in IFC4 schema.

In the process of fire-related data preparation, since it is unethical and immoral to obtain real sensor data through field experiments, computer simulation is chosen to obtain fire-related data to reflect real fire conditions. Fire Dynamic Simulation (FDS) relies on a fluid dynamics model to simulate fire-driven fluid flow [49], and the simulation results are

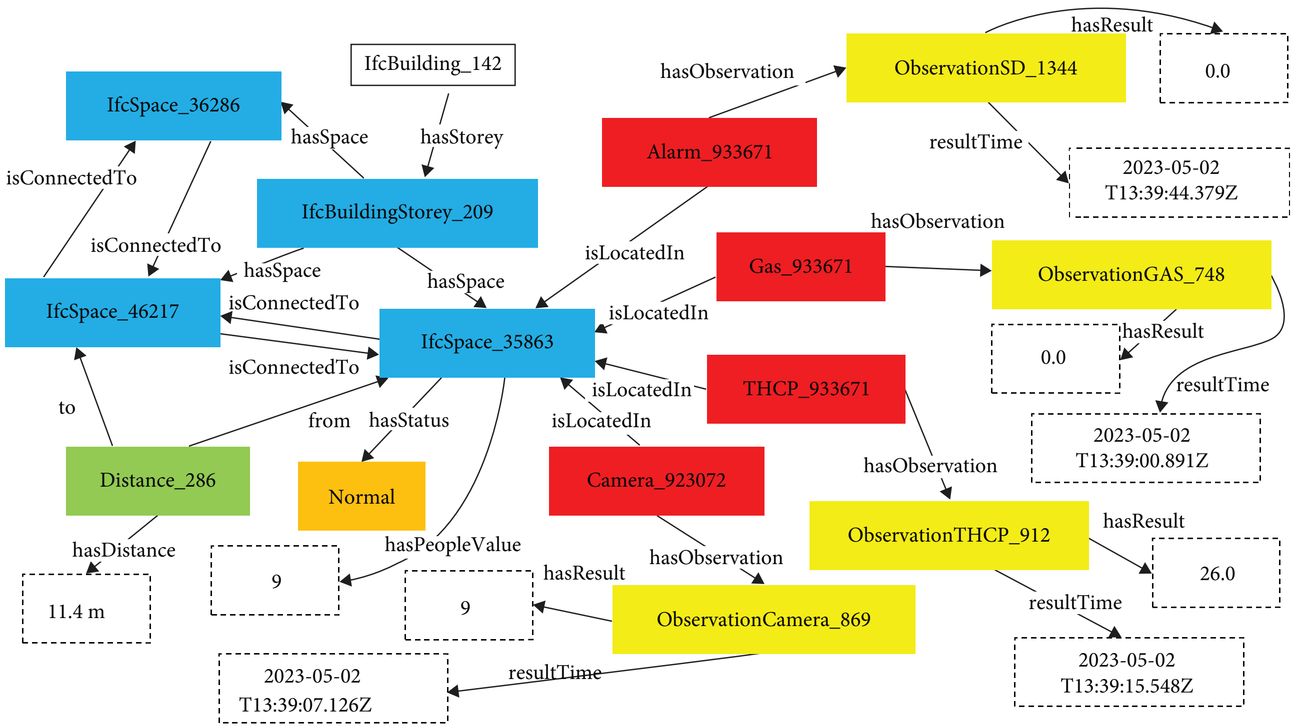


Figure 11: Part of the knowledge graph constructed.

collected by IoT devices as dynamic monitoring data. This study selects Pyrosim for fire simulation. The parameters are set according to the literature [49]. The ambient temperature is set to 26°C, the relative humidity is set to 40%, and the grid size is set to 0.4 m × 0.4 m × 0.4 m. A circular fire source with a heat release rate increasing at $t2$ is set on a 2 m × 4 m burning surface in the center of a room. The fire source power is defined as $Q = 8{,}000$ kW. Based on the fire source parameters, the fire source rise time is calculated to be $t =$ 413 s, in which the growth coefficient is set to 0.04689. The flame starts from the center point of the combustible surface and propagates outward at a speed of $V0 = 0.01$ m/s. The fire extinguishing coefficient of the burning surface is set to 0.1 (m^2/kg)/s. It is assumed that the fire occurred at 13 : 38 : 00.

5.2. Information Extraction and Knowledge Graph Construction. Following the methods outlined in Section 4, the IFC file exported from the BIM model is processed to extract entities defined by the FE ontology. These entities and their relationships are then recorded in the knowledge graph. It is worth noting that temperature sensors and CO sensors are not defined in the IFC data schema. Therefore, during the creation of the BIM model, custom devices are initially used as placeholders, and then device entities and their location information are manually added to the knowledge graph during fire simulation modeling. Subsequently, fire simulations are performed, and sensor observation values are recorded in the instantiated ontology. A partial view of the completed knowledge graph is shown in Figure 11. Figure 11 displays entity instances of all classes, along with examples of object properties and data properties possessed by these entities. The illustration primarily focuses on entities and relationships related to IfcSpace_35863, showcasing the storey it is located on (IfcBuildingStorey_209), evacuation distance to the adjacent space entity IfcSpace_46217 (Distance_286, evacuation distance 11.4 m), the device entities it contains (Alarm_933671, Gas_933671, THCP_933671), the observation values for each device entity, the number of people in IfcSpace_35863, and the safety level of the space. The original data results are stored in Protégé software in the form of a list of triples.

In this study, FDS is applied to simulate fire development. The fire simulation results at each time step are considered as the observation results of IoT devices distributed throughout the building at that moment. The status of spaces is evaluated based on the rules described in Section 4.3. As the fire progresses, the automatic update method of the knowledge graph is shown in Algorithm 1.

In subsequent applications, data from the knowledge graph can be queried using SPARQL queries, as shown in Table 4. The query rules presented in the table can be used for various purposes. For example:

(i) Q1: to compare the results with the BIM model to verify the correctness of ontology instance extraction and population.
(ii) Q2 and Q3: to retrieve the location of device entities and space entities.
(iii) Q4 and Q5: to establish a topological connectivity map within the building, which can be used for evacuation path planning.
(iv) Q6: to determine if there are still people awaiting for evacuation in a space.

Initialize the knowledge graph.
Initialize the time step $t_1, t_2, \ldots, T$.
Initialize the number of IoT devices N.
for $t = t_1, t_2, \ldots, T$ do:
    Retrieve current observation results and fill in the knowledge graph.
    for $n = 1, \ldots, N$ do:
        Check if the result is within the danger range.
        if True do:
            Update the status and evacuation distance of the related space.
        end if
    end for
end for

Algorithm 1: Automatic update method of the knowledge graph.

Table 4: Ontology information retrieval rules.

No.	Query	Aim

Q1	SELECT (COUNT(?space) as ?count)
	WHERE {
	?space rdf:type fe:Space .
	}
		Retrieving the number of space instances in the knowledge graph

Q2	SELECT ?space ?device WHERE {
	?space rdf:type fe:Room .
	?device rdf:type fe:Sensor .
	?device fe:isLocatedIn ?space
	}
		Retrieving the locations of device instances

Q3	SELECT ?storey ?space WHERE {
	?storey rdf:type fe:BuildingStorey .
	?space rdf:type fe:Space .
	?storey fe:hasSpace ?space
	}
		Retrieve the space groups located in a storey

Q4	SELECT ?space_1 ?space_2 ?data
	WHERE {
	?space_1 rdf:type fe:Room .
	?space_2 rdf:type fe:Room .
	?distance rdf:type fe:Distance .
	?distance fe:from ?space_1 .
	?distance fe:to ?space_2 .
	?distance fe:hasDistance ?data
	}
		Retrieving the connectivity relationships between space instances

Q5	SELECT ?space_1 ?space_2 ?data
	WHERE {
	?space_1 rdf:type fe:Room .
	?space_2 rdf:type fe:Room .
	?distance rdf:type fe:Distance .
	?distance fe:from ?space_1 .
	?distance fe:to ?space_2 .
	?distance fe:hasDistance ?data
	}
		Retrieving the evacuation distances between space instances

Q6	SELECT ?space ?data WHERE {
	?space rdf:type fe:Room .
	?space fe:hasPeopleValue ?data
	}
		Retrieving the number of unevacuated people in space instances

Q7	SELECT ?device ?data ?time WHERE {
	?device rdf:type fe:Sensor .
	?device fe:hasObservation ?obs .
	?obs fe:hasResult ?data .
	?obs fe:resultTime ?time
	}
		Retrieving the observation values and observation times of device instances

Table 5: Comparison of two evacuation strategies.

Strategy	Description	Knowledge of spatial relationships	Knowledge of fire development	Safety
1. Nearest evacuation	Preplanned path on evacuation plan	No	No	Potentially hazardous
2. Evacuation based on FE ontology	Dynamic path considering changes of environment	Yes	Yes	Safe

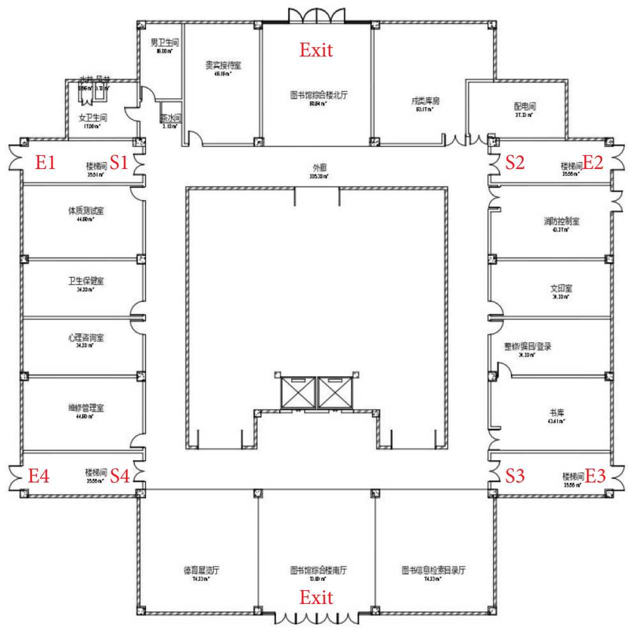


Figure 12: Floor plan of the library.

(v) Q7: to retrieve observation values of devices and update the topological connectivity map based on the criteria outlined in Section 3.2.2.

5.3. Dynamic Evacuation Path Planning. When there is insufficient understanding of the building environment and the evolving fire situation, individuals tend to follow the path displayed on evacuation plan and choose the nearest exit for evacuation [11, 13]. The semantic approach based on FE ontology proposed in this article provides comprehensive knowledge of the evacuation process, assisting individuals in making more rational and effective evacuation decisions, thereby enhancing evacuation efficiency and maximizing personnel safety. Therefore, in this section, the evacuation strategy based on the nearest exit and that based on FE ontology are compared to validate the effectiveness of the semantic approach proposed in this article. The comparison of the two evacuation strategies in terms of knowledge acquisition and safety is presented in Table 5.

The floor plan of the library, as shown in Figure 12, includes two main entrances/exits. Additionally, within each of the four stairwells, labeled S1–S4, there are emergency exits E1–E4 for personnel to use in case of emergencies. During a fire incident, individuals on the first floor can evacuate through the main entrances/exits, while those on upper floors can evacuate to the first floor via the stairs and then exit through the corresponding emergency exits, bypassing the corridors on the first floor. This enhances the efficiency of evacuation. Therefore, for fire evacuation path planning for individuals on upper floors, the starting point is defined as a space with the data property "hasPeopleValue" value not equal to zero, and the endpoints are set as the collection (E1, E2, E3, and E4). Strategy 1 selects the endpoint closest to the starting point from the collection, while Strategy 2 utilizes the Dijkstra algorithm for path planning based on information extracted from the knowledge graph, taking into account changes in space accessibility due to the development of the fire. An example of the fourth-floor space IfcSpace_35344 is taken to compare evacuation paths of two strategies.

Under the parameter settings described in Section 3.1, the fire simulation results for the first floor of the building are shown in Figure 13(a). Using the SPARQL queries from Table 4, at 13 : 38 : 45.514Z, the smoke alarm device "Alarm_854929" located in the space "IfcSpace_317" exceeded the alarm threshold and triggered the fire alarm. At this time, the image recognition results from the camera "Camera_922610" in the fourth-floor space "IfcSpace_35344" indicated that there were a total of eight people waiting to evacuate. Using the queries from Table 4, specifically Q4 and Q5, a weighted network graph of accessible areas is established. In this graph, the nodes represent space entities, and the weights are the evacuation distances. The original

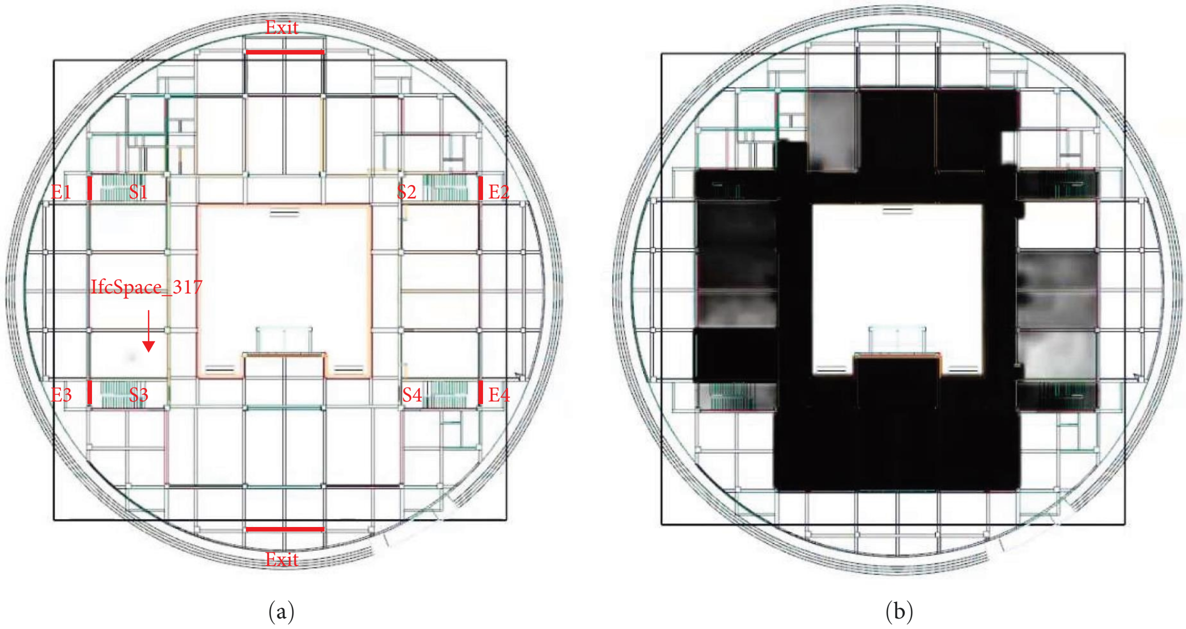


FIGURE 13: Fire simulation results of the moment of alarm (a) and 400 s after the occurrence of fire (b).

connectivity data are stored as a list of triples. A visual topological connectivity graph can be established in tools like Neo4j. The network has been simplified, as shown in Figure 14(a). Since the goal of path planning is to find the nearest exit, paths are considered as unidirectional, leading from rooms to corridors and staircases. It is not assumed that individuals would move from corridors or staircases into other rooms. Therefore, in the network diagram, other rooms have been omitted except for the starting room and staircases. As the fire evolves, the knowledge graph will update accordingly, thereby yielding dynamic weighted network graph. Some key updates are shown in Figure 14. At the alarming point 13 : 38 : 45, there are no impassable danger zones in the building. At this moment, the path planning results of Strategy 1 based on the nearest exit and Strategy 2 based on the knowledge graph are the same. Starting from "IfcSpace_35344," the path planning result, as shown in Table 6, indicates that individuals should evacuate from the public rest area through staircase "S2" to exit "E2," covering a total evacuation distance of 23.5 m.

After 400 s since the occurrence of the fire, at 13 : 44 : 40.031Z, the fire simulation results are shown in Figure 13(b). At this point, based on the results retrieved using Q7 and the rules described in Section 3.2.2, staircases S1, S2, and S4 are all in a dangerous state. Therefore, the evacuation distance values for the spaces connected to these staircases are updated to 1,000 m. Part of the updated network of passable areas is shown in Figure 14(d). At this time, if the image recognition results from camera "Camera_922610" are not equal to 0, indicating that there are still people awaiting evacuation in space "IfcSpace_35344", a new path planning process is initiated.

At this point, according to Strategy 1, the evacuation path for individuals is shown in Table 7. The individuals still follow the preplanned path on evacuation plan and choose to evacuate from the nearest staircase, S2. Upon reaching the staircase space IfcSpace_17043 on the second floor, they perceive that the staircase leading to exit E2 is impassable. They then opt to divert their path through IfcSpace_17559 and IfcSpace_17301 to reach exit IfcSpace_8586 via staircase S4, increasing the total evacuation distance to 70.1 m. Additionally, individuals may face potential risks due to delayed decision-making and the rapid development of the fire. In contrast, according to Strategy 2, the evacuation path for individuals is shown in Table 8. With knowledge provided by the knowledge graph about space connectivity and fire development, Strategy 2, when applying the Dijkstra algorithm for path planning, is aware that staircases S1, S2, and S4 are severely affected by the fire and impassable. Therefore, the planned path guides individuals directly choose staircase S3 for evacuation. This not only shortens the total evacuation distance by 6.8%, but also guides individuals to minimize exposure to fire hazards. According to Strategy 2, the path planning results under different fire development scenarios are shown in Figure 15. The path planning results can be communicated to individuals through guidance from management personnel or through signage distributed throughout the building, as well as feedback from intelligent devices.

6. Conclusions

In conclusion, the main contribution of this study to the field of fire evacuation is introducing a comprehensive, cross-source data management and analysis framework to support evacuation path planning and emergency response decisions in fire scenarios. By developing a specialized FE ontology and knowledge graph of the fire evacuation domain, this study achieves several key objectives.

First, it successfully integrates data from multiple sources, including BIM models, IoT devices, image recognition, and fire simulations, enabling a more holistic understanding of

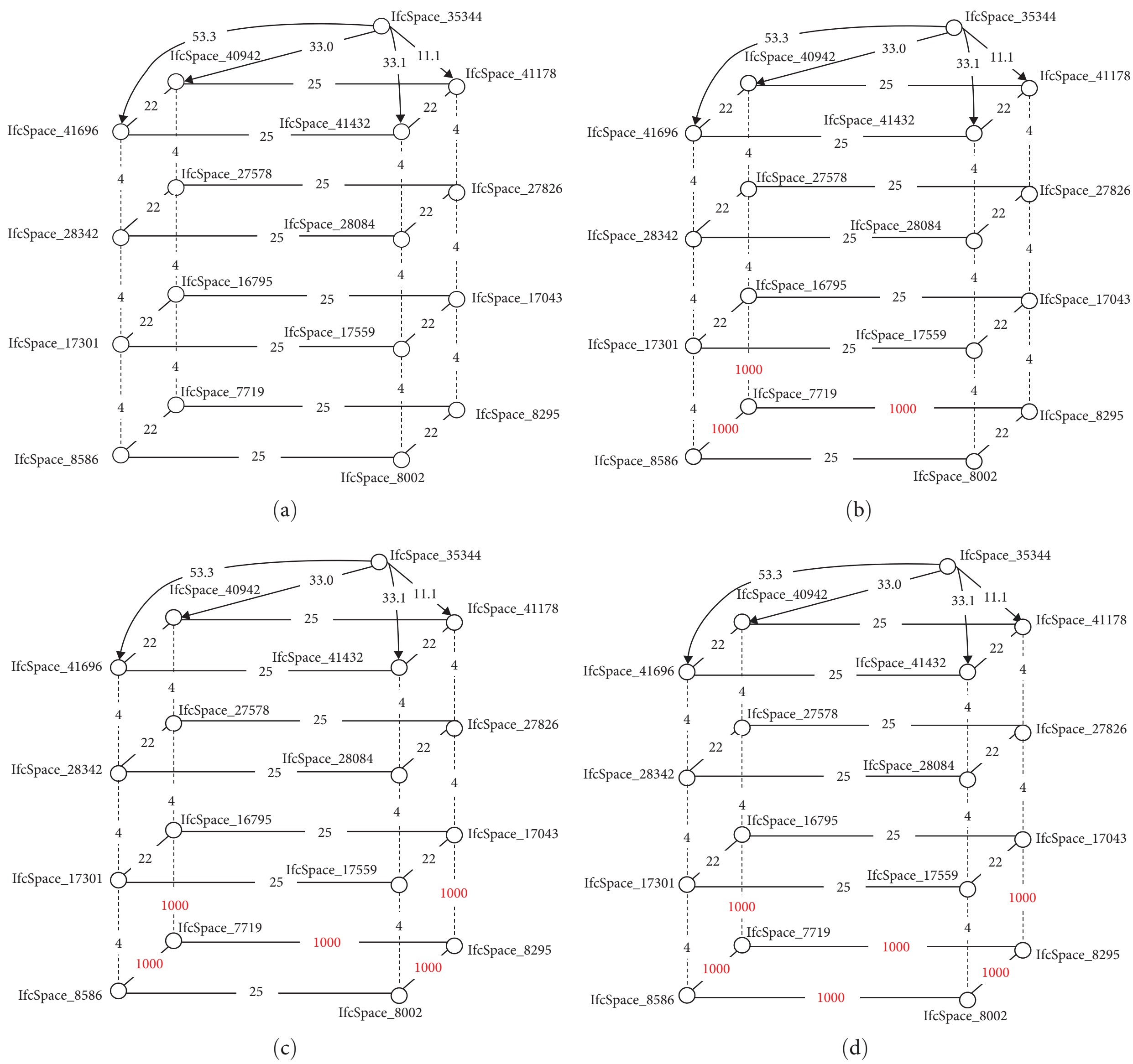


Figure 14: Updates of weighted connection graph of spaces of the moment of alarm 13 : 38 : 45 (a), and 269.5 s (b), 339.3 s (c), 400 s (d) after the occurrence of fire.

Table 6: Result of path planning at 13 : 38 : 45.

Start	End	Distance (m)	Path	Time (ms)
IfcSpace_35344	IfcSpace_8295	23.1	["IfcSpace_35344", "IfcSpace_41178", "IfcSpace_27826", "IfcSpace_17043, "IfcSpace_8295"]	86

Table 7: Result of path planning by Strategy 1 at 13 : 44 : 40.

Start	End	Distance (m)	Path	Time (ms)
IfcSpace_35344	IfcSpace_8586	70.1	["IfcSpace_35344", "IfcSpace_41178", "IfcSpace_27826", "IfcSpace_17043", "IfcSpace_17559", "IfcSpace_17301", "IfcSpace_8586"]	—

Table 8: Result of path planning by Strategy 2 at 13 : 44 : 40.

Start	End	Distance (m)	Path	Time (ms)
IfcSpace_35344	IfcSpace_8586	65.3	["IfcSpace_35344", "IfcSpace_41696", "IfcSpace_28342", "IfcSpace_17301", "IfcSpace_8586"]	93

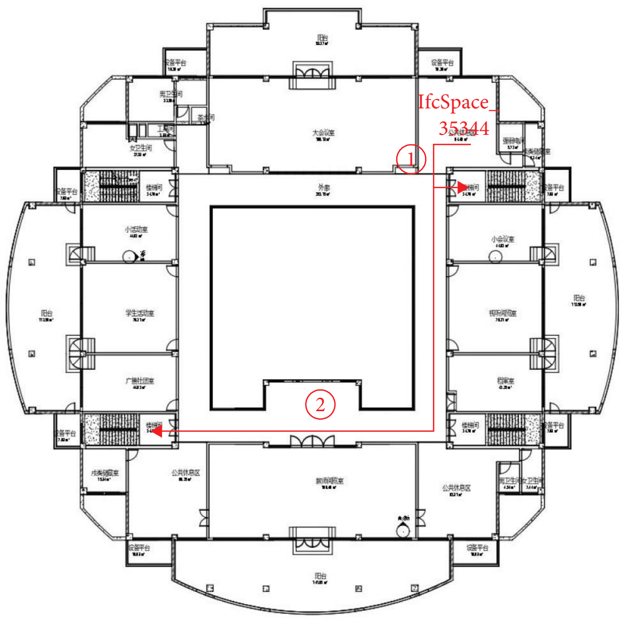


Figure 15: Evacuation path of the moment of alarm 13 : 38 :45 (1) and 400 s after the occurrence of fire 13 : 44 : 40 (2).

building environmental characteristics, evacuation path accessibility, and fire development. This integration provides substantial support for effective emergency responses and the planning of evacuation routes.

Second, the study thoroughly details methods for data extraction and the acquisition of fire-related information during ontology entity population and knowledge graph construction. This step establishes a comprehensive, consistent, and semantically rich data representation and management framework for the field of fire evacuation. Critical architectural data are extracted from BIM models, while real-time IoT device observations are enriched using simulation and image recognition techniques, enhancing the knowledge graph's content.

Moreover, the feasibility of the proposed FE ontology framework is demonstrated by applying it to a fire scenario path planning example within a specific building. Compared to the strategy of choosing the nearest exit, the semantic method based on FE ontology provides knowledge about the building environment and fire development, allowing for dynamic updates of real-time evacuation path, ensuring the safety of individuals undergoing evacuation. This validation provides concrete evidence of the method's practical value in real emergency situations.

However, it is important to acknowledge certain limitations and future research directions. First, the ontology and knowledge graph developed in this study primarily cater to specific building environments and require further expansion and generalization to accommodate different building types. Second, data processing methods, particularly in the realm of image recognition, can benefit from more accurate algorithms to yield more precise results. Finally, considering the impact of human factors such as movement speed and congestion on evacuations is crucial, and future research should incorporate human-related information to better model evacuation behavior in fire scenarios.

Data Availability

The data used to support the findings of this study are available from the first author upon request by email bo.pang.sjtu@gmail.com.

Conflicts of Interest

The authors declare that there is no conflict of interest regarding the publication of this article.

References

[1] U.S. EPA, "Indoor air quality," 2018, *Report on the Environment*, https://www.epa.gov/report-environment/indoor-air-quality.

[2] E. Young and B. Aguirre, "PrioritEvac: an agent-based model (ABM) for examining social factors of building fire evacuation," *Information Systems Frontiers*, vol. 23, no. 5, pp. 1083–1096, 2021.

[3] A. O. Arewa, A. Ahmed, D. J. Edwards, and C. Nwankwo, "Fire safety in high-rise buildings: is the stay-put tactic a misjudgement or magnificent strategy?" *Buildings*, vol. 11, no. 8, pp. 1–16, 2021.

[4] A. A. Stec, K. Dickens, J. L. J. Barnes, and C. Bedford, "Environmental contamination following the Grenfell tower fire," *Chemosphere*, vol. 226, pp. 576–586, 2019.

[5] NFPA research, "Fire loss in the United States," 2023, U.S. NFPA, https://www.nfpa.org/education-and-research/research/nfpa-research/fire-statistical-reports/fire-loss-in-the-united-states.

[6] "Regular press conference for January of the Ministry of Emergency Management of China," State Counc Inf of People's Repub China, 2022.

[7] S. Wang, W. Wang, K. Wang, and S. Shih, "Applying building information modeling to support fire safety management," *Automation in Construction*, vol. 59, pp. 158–167, 2015.

[8] Y. Wan, T. Hu, B. He, and B. Yang, "A path optimization method of three-dimensional dynamic pedestrian evacuation," Fire Safety Science, 2021.

[9] G. Hofinger, R. Zinke, and L. Künzer, "Human factors in evacuation simulation, planning, and guidance," *Transportation Research Procedia*, vol. 2, pp. 603–611, 2014.

[10] Y. Wang, J. Zheng, and B. Ding, "Fire protection monitoring system for power cabin in utility tunnel based on IoT technology," in *2021 IEEE 5th Information Technology, Networking, Electronic and Automation Control Conference (ITNEC)*, pp. 1366–1371, IEEE, Xi'an, China, 2021.

[11] ISO 23601, "Safety identification—escape and evacuation plan signs," International Organization for Standard, 2020.

[12] A. M. Kassim, M. M. Roslan, S. Sahak et al., "Performance evaluation of intelligent fire alarm system with multi-data fusion sensor by using IoT platform," in *Proceedings of the Symposium on Intelligent Manufacturing and Mechatronics (SympoSIMM 2021)*, pp. 199–209, Springer, Singapore, 2021.

[13] ISO 20414, "Fire safety engineering—verification and validation protocol for building fire evacuation models," International Organization for Standard, 2020.

[14] Y. Li, J. Yi, X. Zhu, Z. Wang, and F. Xu, "Developing a fire monitoring and control system based on IoT," *Proceedings of the 2016 2nd International Conference on Artificial Intelligence and Industrial Engineering (AIIE 2016)*, vol. 133, pp. 174–178, 2016.

[15] J. Yuan, R. Chen, and P. Yu, "Application of navigation grid corner point algorithm in virtual reality simulation images of indoor fire evacuation," *Internet of Things*, vol. 22, Article ID 100716, 2023.

[16] Q. Sun and Y. Turkan, "A BIM-based simulation framework for fire safety management and investigation of the critical factors affecting human evacuation performance," *Advanced Engineering Informatics*, vol. 44, Article ID 101093, 2020.

[17] H. Fang, S. Lo, and J. T. Y. Lo, "Building fire evacuation: an IoT-aided perspective in the 5G era," *Buildings*, vol. 11, no. 12, Article ID 643, 2021.

[18] H. Chen, L. Hou, G. Zhang, and S. Moon, "Development of BIM, IoT and AR/VR technologies for fire safety and upskilling," *Automation in Construction*, vol. 125, Article ID 103631, 2021.

[19] N. Li, B. Becerik-Gerber, B. Krishnamachari, and L. Soibelman, "A BIM centered indoor localization algorithm to support building fire emergency response operations," *Automation in Construction*, vol. 42, pp. 78–89, 2014.

[20] X.-S. Chen, C.-C. Liu, and I.-C. Wu, "A BIM-based visualization and warning system for fire rescue," *Advanced Engineering Informatics*, vol. 37, pp. 42–53, 2018.

[21] N. F. Noy and D. L. McGuinness, ""Ontology development 101: a guide to creating your first ontology," 2001.

[22] T. Hartmann and A. Trappey, "Philosophical and methodological foundations with examples from civil and construction engineering," *Advanced Engineering Informatics*, vol. 4, Article ID 100020, 2020.

[23] J. Neto, A. J. D. N. Morais, R. Gonçalves, and A. L. Coelho, "An ontology for fire building evacuation," in *Sixth International Congress on Information and Communication Technology: ICICT 2021*, Springer, Singapore, 2021.

[24] W3C, "SPARQL 1.1 query language[EB/OL]," 2022, https://www.w3.org/TR/rdf-sparql-query/.

[25] W3C, "RDF 1.1 primer[EB/OL]," 2022, http://zh.transwiki.org/cn/rdfprimer.htm.

[26] X. Li, G. Liu, A. Ling et al., "Building a practical ontology for emergency response systems," in *International Conference on Computer Science and Software Engineering*, vol. 4, pp. 222–225,2008, IEEE, Wuhan, China, 2008.

[27] A. Malizia, T. Onorati, P. Diaz, I. Aedo, and F. Astorga-Paliza, "SEMA4A: an ontology for emergency notification systems accessibility," *Expert Systems with Applications*, vol. 37, no. 4, pp. 3380–3391, 2010.

[28] N. Nuo, N. Kubota, and J. Botzheim, "Building ontology for fire emergency planning and support," *E-Journal of Advanced Maintenance*, vol. 8-2, pp. 13–22, 2016.

[29] Ü. Öztürk, M. Akdağ, and T. Ayabakan, "A review of path planning algorithms in maritime autonomous surface ships: navigation safety perspective," *Ocean Engineering*, vol. 25, no. 1, Article ID 111010, 2022.

[30] H. Liao, Y. Zhu, and Z. Chen, "A fire disaster dynamic evacuation indicating system based on the Dijkstra algorithm," *Journal of Safety and Environment*, vol. 21, no. 4, pp. 1676–1683, 2021.

[31] X. Huang, J. Jin, Z. Lin, L. Che, and J. Liu, "Dynamic evacuation path planning for fire disaster of deep underground space based on A^* algorithm," *Journal of Beijing University of Technology*, vol. 47, no. 7, pp. 702–709, 2021.

[32] P.-J. Barre, J.-Y. Dieulot, R. Bearee, and A. Bouzidi, "A heuristic path-planning method for enhancing machine-tool contour-following," *Journal of Systems Science and Systems Engineering*, vol. 14, no. 1, pp. 85–96, 2005.

[33] J. Wang, W. Chi, C. Li, C. Wang, and M. Q. H. Meng, "Neural RRT*: learning-based optimal path planning," *IEEE Transactions on Automation Science and Engineering*, vol. 17, no. 4, pp. 1748–1758, 2020.

[34] W. Mao, S. Li, X. Xie, X. Yang, and J. Nie, "Global path planning of mobile robot based on adaptive mechanism improved ant colony algorithm," *Control and Decision*, vol. 38, no. 9, pp. 2520–2528, 2023.

[35] C. Li, Y. Li, F. Huo, and Q. Zhang, "Fire evacuation model based on dynamic coupling of FDS and cellular automata," *Application Research of Computers*, vol. 39, no. 9, 2022.

[36] C. Huang, Y. Chiang, and F. Tsai, "An ontology integrating the open standards of city models and internet of things for smart-city applications member," *IEEE*, vol. 9, no. 20, 2022.

[37] L. Y. Ding, B. T. Zhong, S. Wu, and H. B. Luo, "Construction risk knowledge management in BIM using ontology and semantic web technology," *Safety Science*, vol. 87, pp. 202–213, 2016.

[38] K. Janowicz, A. Haller, S. J. D. Cox, D. Le Phuo, and M. Lefrançois, "SOSA: a lightweight ontology for sensors, observations, samples, and actuators," *Web Semantics: Science, Services and Agents on the World Wide Web*, vol. 56, pp. 1–10, 2019.

[39] S. Ji, S. Pan, E. Cambria, P. Marttinen, and P. S. Yu, "A survey on knowledge graphs: representation, acquisition, and applications," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 33, no. 2, pp. 494–514, 2022.

[40] D. Zhou, B. Zhou, Z. Zheng et al., "Enhancing knowledge graph generation with ontology reshaping-bosch case," in *The Semantic Web: ESWC. 2022 Satellite Events(ESWC 2022)*, vol. 13384, Springer, Cham, Lecture Notes in Computer Science.

[41] Y. Wang, J. Zhu, and B. Wei, "Domestic and international mainstream BIM software application and comparison study," *Journal of Physics: Conference Series*, vol. 2185, Article ID 012088, 2022.

[42] M. F. Muller, A. Garbers, F. Esmanioto, N. Huber, E. R. Loures, and O. Canciglieri Jr, "Data interoperability assessment though IFC for BIM in structural design-a five-year gap analysis," *Journal of Civil Engineering and Management*, vol. 23, no. 7, pp. 943–954, 2017.

[43] G. Ralf, *The Future of IFC: Enabling Partial BIM Model Exchange*, Upfront E-zine, 2020.

[44] "BuildingSMART International," 2022, https://ifc43-docs.standards.buildingsmart.org/.

[45] C. Wang, J. Liu, Z. Pan, and J. Shi, "An intelligent aided code compliance checking approach for passive fire protection based on BIM and ontology," *Fire Science and Technology*, vol. 42, no. 1, pp. 64–69, 2023.

[46] A. Vora and V. Chilaka, "FCHD: fast and accurate head detection in crowded scenes," *Computer Vision and Pattern Recognition*, arXiv: 1809.08766.

[47] M. Hasofera and I. Thomas, "Analysis of fatalities and injuries in building fire statistics," *Fire Safety Journal*, vol. 41, no. 1, pp. 2–14, 2006.

[48] Y. Ding, X. He, Q. Zhu, H. Lin, and M. Hu, "A dynamic optimization method of indoor fire evacuation route based on real-time situation awareness," *Acta Geodaetica et Cartographica Sinica*, vol. 45, no. 12, pp. 1464–1475, 2016.

[49] M.-X. Li, S.-B. Zhu, J.-H. Wang, and Z. Zhou, "Research on fire safety evacuation in a university library in Nanjing," *Procedia Engineering*, vol. 211, pp. 372–378, 2018.