

Article

Automatic Generation of Guidance for Indoor Navigation at Metro Stations

Jacek Bernard Marciniak *  and Bartosz Wiktorzak 

Faculty of Geodesy and Cartography, Warsaw University of Technology, Pl. Politechniki 1, 00-661 Warsaw, Poland; bwiktorzak@gmail.com

* Correspondence: jacek.marciniak@pw.edu.pl

Abstract: This article delves into the advancements in indoor navigation in metro stations and describes the development and implementation of algorithms for the automatic generation of navigation guidance. The LIFT project at the Warsaw University of Technology serves as a practical example, showcasing a system designed to cater to people with special needs. This article presents a rule-based algorithm that generates navigation directions based on a trade-off between landmark references and spatial references in relation to the user's location. The research uses a spatial data model consisting of three interconnected layers: the transport network, the room topology, and the building topography. The algorithm uses these data in subsequent stages. A defined set of rules generates redundant navigation directions for all potential decision points and then, subsequent rules filter and generalise them. To discuss the details of how the algorithm works, an example route is described in this study and the consequences of applying the selected rules are analysed. Next, a few problems that arose during the testing of the algorithm at Warsaw Metro stations are presented with proposed solutions. The results of the study made it possible to develop a mobile application, which is planned to be available to users by the end of 2024.

Keywords: indoor navigation; indoor guidance; landmarks; turn-by-turn directions; people with special needs; metro



Citation: Marciniak, J.B.; Wiktorzak, B. Automatic Generation of Guidance for Indoor Navigation at Metro Stations. *Appl. Sci.* **2024**, *14*, 10252. <https://doi.org/10.3390/app142210252>

Received: 28 September 2024

Revised: 1 November 2024

Accepted: 4 November 2024

Published: 7 November 2024



Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

In recent decades, there has been significant development in technologies supporting indoor navigation, reflected in the growing interest of both researchers and engineers. Newly built indoor navigation systems must meet the increasing demands of users moving not only in buildings with well-defined systems of rooms and corridors, but also in complex spaces such as large shopping malls, airports, public transport terminals, and metro stations. In such environments, traditional navigation guidance based on turn-by-turn directions is ineffective. This is the reason why researchers have investigated the possibility of navigation guidance based on landmarks.

A landmark is an easily recognisable point in space that is used in navigation. It can be a natural point, such as hills, floodplains, or rivers, or a man-made point, e.g., architectural structures such as churches or towers. The use of landmarks in guidance significantly improves the understanding of the route course in many types of navigation [1,2]. The term is used in indoor navigation to refer to objects that are architectural elements or permanent fixtures of a building that are easy to notice, such as a corridor that diverges to the side, an unusual door decoration, or a hanging painting. In the literature, we can find studies that use landmarks at different stages of indoor navigation, starting with localisation [3]. A more common use of landmarks is to describe the building interior to improve indoor guidance [4,5]. The modelling of the interior space can be based on an ontology that defines object categories as well as their semantical relationships and topology [6,7]. The full ontology supports the issue of how to effectively construct an indoor navigation

direction. As [8] discusses, the key challenge of landmark-based guidance is to select the right landmarks and describe topological relationships to the user's manoeuvres. In later research, a more popular approach was to use hierarchical systems of landmarks and relate them to calculated routes instead of describing all elements of indoor topography [9–11].

Metro stations are key places of public transportation in many metropolises. The effective navigation of passengers to the relevant platforms is essential for ensuring the smooth flow of people and reducing possible confluence. The complex architecture of these facilities, combined with the lack of access to natural light and limited visibility, presents numerous challenges. In these conditions, technology based on landmarks gains particular significance, enabling intuitive guidance by referencing distinctive elements of the environment.

Another challenge to consider is adapting navigation systems to the needs of people with special needs. Modern standards for designing public infrastructure increasingly emphasize accessibility and inclusivity, requiring that navigation instructions be not only precise but also understandable and easily accessible to all users, regardless of their abilities. This does not just come down to designing systems that account for diverse needs, such as voice navigation for the visually impaired or avoiding stairs for people in wheelchairs. Many solutions focus on navigating people with disabilities via reliable voice directions without displaying a map [12,13]. The key point is to understand the actual expectations of users and to consider them in all phases of the navigation system's implementation, starting with analysis, modelling the spatial database, and designing the user interface and user experience.

Despite interesting research results, the landmark-based approach has not become widespread. Fully basing navigation guidance on landmarks is an expensive process that requires data acquisition, data maintenance with frequent updates, and building complex algorithms. A more common approach involves wayfinding with standard graph-based algorithms and automatically generated directions using navigation graphs and calculated routes [14,15].

This approach was implemented in the project 'Development of a prototype system to support the movement of people with special needs inside architectural structures related to rail transport—LIFT', conducted at the Warsaw University of Technology in 2022–2024. The research was funded by the National Centre for Research and Development in Poland and conducted by an interdisciplinary team of researchers, architects, cartographers, and software engineers. The system analysis was preceded by visits to several European cities, such as Barcelona, Brussels, Amsterdam, and Berlin, to gather best practices of passenger information systems and local applications. The requirements of the system were determined following consultation with experts from foundations working with people with disabilities, who understand the real problems and can evaluate the proposed solutions. Preliminary results of the research were presented at the CSUN Assistive Technology Conference [16] and received a positive response. The project will result in a mobile application being made available to Warsaw Metro users by the end of 2024.

This article starts with a comprehensive analysis of the issue of generating navigation guidance in the context of metro stations and other underground spaces. It focuses on discussing contemporary trends in indoor navigation, the challenges of underground navigation, and how modern technologies can contribute to improving the accessibility and efficiency of navigation systems. The analysis is followed by discussion of its outcomes, including the presentation of the algorithm implemented and tested under real-life conditions.

2. Related Work

Indoor navigation systems have been developed for many years now. In the early days of technology, developers focused on positioning technology. Nowadays, an increasing amount of research is being conducted on indoor navigation guidance—how to create appropriate navigation directions that intuitively guide users to their destinations.

The aim of the LIFT project was to create an indoor navigation system for people with special needs. The system was designed to work in buildings at rail transport stations, with particular emphasis on metro stations. In order to build the system, a number of problems had to be addressed in various areas: the development of urban commuter planners, navigation guidance in indoor navigation systems in general, indoor navigation systems for buildings with irregular architecture, and indoor navigation systems for people with special needs.

2.1. Urban Commuter Planners

Commonly used map applications provide routing functions based on public transport. This type of route can be found in Google Maps, Bing Maps, Here Maps, and applications specifically developed for public transport, such as Citymapper. Algorithms have been developed to support multi-modal routing [17,18], and today's systems allow routes to be calculated with transfers both between one type of transport, e.g., between different bus lines, and between different types of transport, e.g., buses and trams. The advantage of these systems is that they work in many cities around the world, and a user accustomed to the application can use them in another city. On the other hand, many cities provide applications for mobile devices that take into account specific conditions and improve the performance of generic systems. While generic systems usually do not consider users with special needs, city-specific apps are built with different assumptions. Some studies, e.g., for Paris [19] or Madrid [20], lay the foundations for commute planners for different types of users, including people with disabilities.

The urban commuter planners focus on planning the optimal route in terms of choosing suitable buses, trams, trains, or underground. However, the issue of pedestrian navigation in areas of terminals or stations is beyond the scope of this study. While some applications include exit names, the challenge remains in finding the right one and presenting information consistent with the physical signs at the station. Due to the lack of a proper description of station exits and the lack of consideration of areas inaccessible to some people, these apps can often provide routes that confuse disabled users in complex buildings.

2.2. Indoor Navigation in Bus Terminals, Railway Stations, and Airports

A critical part of navigating users of public transport is the transfer to another vehicle. The process might be straightforward if it requires a short walk between outdoor stops, but it becomes challenging if it takes place inside a building. In that case, it incorporates indoor navigation. Examples of public transport routes provided by popular applications show a tendency to generate turn-by-turn directions. This type of guidance is often impractical indoors because many people have difficulty orienting themselves with reference to the map. Even applications that include positioning components fail to provide valuable real-time directions due to the low accuracy of determining the user's position.

Research on airport navigation applications [21–23] concludes that one of the most important assumptions to be taken into account at the initial stage of designing the system is that airports are not navigation destinations but rather transitional spaces. The building is accessed from the outside by selecting the appropriate entrance from a number of, sometimes distant, entrances. Once inside the building, the user performs standard tasks such as baggage check-in, going through security, and finding the departure gate. Similar problems are encountered in pedestrian navigation at railway stations [24,25]. In such cases, users go to the ticket office and find the correct platform.

2.3. Indoor Navigation for People with Special Needs

The seemingly simple tasks described in the previous section can pose problems for people who have poorer spatial orientation. Specific conditions inside transit buildings, such as limited visibility, distracting ambient noise, and crowds of people, make it difficult for some users to move efficiently. Reaching a building through the most convenient entrance and then navigating around to proceed to a platform at a station or a gate at

an airport is particularly difficult for people with disabilities. Typical problems faced by visually impaired people at metro stations and approaches to solving them are presented in the literature [26,27]. In recent years, artificial intelligence methods have been used to aid indoor navigation. In addition to improving position accuracy [28–30], deep learning and reinforcement learning methods can facilitate the movement of visually impaired people through obstacle detection and landmark identification [31,32] or spatial analysis for localisation [33].

Indoor navigation systems for wheelchair users have been the subject of research on requirements and general conditions of navigation [34–36]. Similarly to the research on the navigation of visually impaired people, deep learning and reinforcement learning methods can be used for the identification and avoidance of obstacles [37] and for recognising landmarks [38].

The conclusions of these studies clearly indicate that systems should be designed with the specific type of user in mind, and paying attention to the specific limitations they have. User expectations must be considered not only during the creation of the navigation guidance but also at earlier stages of building the system, starting with the modelling of the spatial data.

3. Materials and Methods

3.1. The Key Assumptions of the System

Current indoor positioning algorithms that can be implemented in public buildings or metro stations have poor accuracy. The most popular systems based on Bluetooth or Wi-Fi positioning have accuracies of a few meters in real-world conditions [39,40]. The accuracy of 1 m, as declared by some researchers, has not been confirmed by the observations of real users testing applications on their own mobile phones. Given that, firstly, it is difficult to obtain an accurate position indoors, especially in complex transportation hubs, and secondly, that the system is built for people with special needs, the authors decided to focus on the research on algorithms that generate navigation guidance. By design, the system can be used without a positioning system. The user finds a route, as in a typical urban commuter planner, and receives directions to follow. The developed algorithms can be used alone or integrated into a system with a positioning module, providing additional improvement.

The analysis confirmed by tests with people with disabilities showed that without a positioning system, a kind of spatial reference is necessary anyway, without which the user may accidentally leave the route and not be able to return to it. This need is addressed in the LIFT system by appropriately designed navigation guidance. In addition to a schematic presentation of the route on a building plan, each direction displays a photo taken at the decision point. Such pictures facilitate the identification of route points and reassure the user that they are following the route.

3.2. Space Topology Modelling

The problem of describing indoor space as a graph was analysed in a document published by the OGC (Open Geospatial Consortium) in 2010 [41] and, based on the findings, the open standard IndoorGML was proposed in 2014 [42]. It defines general principles for describing the interior of a building as a graph, in terms of modelling spatial data for navigation. The basis of the model is a graph describing geometrical and topological relationships between rooms and/or structural elements, providing more complex information, e.g., to move between rooms, one needs to use specific doors. Multilayer representation of the space can model different types of relationships, such as adjacency of objects or ‘navigability’, i.e., the physical connections that allow movement between rooms [43].

As shown in studies [44,45], room topology graphs can be directly used for routing in buildings, and the route can be represented as a list of rooms. In addition, it was shown [46] that such a graph can be combined with a transport network outside the building to determine routes between buildings.

3.3. Route Geometry

The description of the route as a list of consecutive rooms is easy to implement. However, it may not be sufficient to convey a clear message to the user as to how the route goes. A person is accustomed to the presentation of a mapped route as a linear trajectory along which they should navigate. In order to obtain such a trajectory, it is necessary to model the geometries of the paths of movement through the building. This problem was the focus of a study [47] in which the authors analysed the positioning of various objects in rooms that represented obstacles during movement and, on that basis, determined the geometries of potential paths. A different approach to automatically determine the geometries of paths was used in another study [48]; the space was divided into logical parts and the geometry determined automatically by triangulating polygons representing the surfaces of the rooms and smoothing the resulting lines.

Once the route has been calculated, the system should present it to the user, usually as a line on the map that represents the expected trajectory of the user's movement. To create this geometry, the algorithms use a transport network to represent the possible paths of movement of objects. The route is calculated with a graph related to the transport network, and the final route geometry is obtained by connecting the geometries associated with the edges of the graph.

In car navigation systems, which users are familiar with, there is a road network that gives a good indication of the possible positions of the vehicle. Cars travel along streets and, except in exceptional situations, change direction when they can turn into another street. The road network can be modelled as a graph such that the edges represent road segments and the vertices represent intersections. In the case of indoor navigation, it is not possible to indicate an analogous road network. If the building consisted only of corridors, their axes could be used. However, determining the geometry in a natural transport network is problematic if the route runs through other rooms, open spaces, halls, etc.

The creation of a path network in a building can be partly automated, but even then, it is good practice to analyse the routing of potential path trajectories and possibly modify the geometry manually. A previous article [49] presented typical cases of path geometry modelling based on the assumption that paths connect characteristic points of a room by straight lines, each with the other. A more comprehensive approach can be found in [50], where the author provides complete guidance on how to model navigation graphs. An analogous approach, supplemented with additional conditions, was used in [51].

3.4. Landmarks and Turn-by-Turn Directions

As studies [4,52,53] show, turn-by-turn navigation does not work well indoors. Pedestrians describe the route differently—they do not use manoeuvres to be executed at decision points but use more general cues specifying directions of movement. In addition, a direction such as 'turn left/right' can be difficult to interpret due to the inaccuracy of position and the ease with which the body can be reoriented in space.

A more intuitive approach is based on the use of landmarks [54–56]. Such landmarks are defined as easily recognisable objects, e.g., mountain peaks or characteristic buildings. In indoor navigation, these can be unchanging, easily identifiable elements of the building façade, e.g., ornaments, or also paintings hanging on the wall. The guidance provided during navigation refers to such points so that the information message is unambiguous, and the user understands how to perform subsequent manoeuvres. The main problem with landmark-based guidance is the cost of acquiring and updating data. Some landmarks are permanent fixtures in buildings, such as a fire extinguisher; others are slightly more variable, such as a painting that can be taken down or rehung in another location; and others are dynamic, such as tables or chairs.

3.5. Multilayer Data Model

An effective algorithm framework based on two graphs, one representing the room topology and the other the transport network, and maintaining links between them is

discussed in [57]. In the LIFT project, the spatial data model consists of three connected layers: the transport network, the room topology, and the building topography. The meaning of the layers in the algorithm is summarised in Table 1.

Table 1. Data layers used in the algorithm for the automatic generation of indoor guidance.

Layer	Data Structure	Objects	Use of the Algorithms
Transport network	Graph	Edges represent segments of pedestrian paths; nodes represent connections between path segments.	This graph is primarily used to calculate the route and create the route geometry. In the user tracking process during navigation, the geometries are used in map-matching.
Room topology	Graph	Nodes represent rooms; edges represent connectivity between rooms.	Once the route is calculated, these data allow the enrichment of directions by adding the overall information about the building, i.e., room changes.
Topography	Spatial objects	Floors, rooms, stairs, lifts, etc., represent parts of the building; landmarks represent any objects intentionally selected.	Provide additional information that can be used in creating comprehensive directions, such as room numbers, names, accessibility, etc. Landmarks are used to add spatial reference to navigation directions. Points of interest are used in the search function.

This model defines the responsibilities of each layer and simplifies the process of generating complex navigation directions. The process starts with finding the optimal route with the Dijkstra algorithm. At this stage, the transport network is used. The path network is represented as a graph with associated geometries of path segments. Such a graph is often referred to in the literature as a navigation graph. Some authors, like [50], define the navigation graph as a graph whose edges represent paths and whose nodes represent decision points, landmarks, or locations where information must be transmitted or presented by the system. This definition would cover all data used in the navigation process. To avoid confusion, the authors avoid using the term ‘navigation graph’ and refer to the specific layers of the data model.

Once the route is calculated and its geometry is created with path segment polylines, the system generates navigation guidance. In this process, the topology graph is used. This makes it easy to determine at which point a change in room, an exit to a corridor, the ascent of a staircase, or other events occur that are easily interpreted by a user. The geometry of the route trajectory is useful for creating turn-by-turn directions, which in general are not optimal but in some cases, when the position of the user is known, can be helpful.

To create comprehensive navigation guidance, additional information should be added. Typical information covers names or numbers of rooms, some distinctive features of the interior, as well as landmark descriptions. These data are retrieved from the topography layer.

3.6. Characteristics of Metro Station Interiors

Metro stations have a unique construction that sets them apart from other buildings. Understanding the characteristic structure of the Warsaw Metro station is important for modelling the data.

Stations typically feature multiple entrances located on various sides of a street, without the possibility of assigning a single address representing the station in a search engine. The shape of stations is generally linear, consisting primarily of corridors. Some stations feature an intermediate level called a mezzanine floor, situated between the platform and the ground surface. The mezzanine can be simple and located on one side of the station, or it can be extensive, connecting to major transfer stations and incorporating small galleries and service points. Some metro stations have connected platforms with a single access, while others have two separate platforms for trains travelling in opposite directions. Metro stations typically feature numerous points of interest, including turnstiles, platforms, and ticket machines. A 3D view based on the LIDAR point cloud for one of the Warsaw Metro stations is presented in Figure 1.

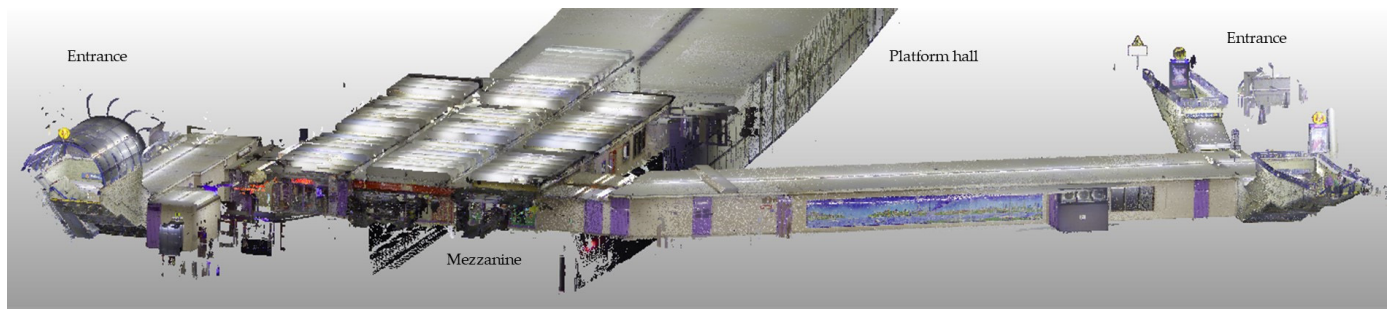


Figure 1. LIDAR point cloud of the southern part of metro station 'Politechnika'.

3.7. Spatial Data Model

The model of spatial data used in the LIFT project, with clear division into three layers, was based on earlier work of the research team [57–59] with modifications specific to metro stations. The classes of the model are presented in Figure 2.

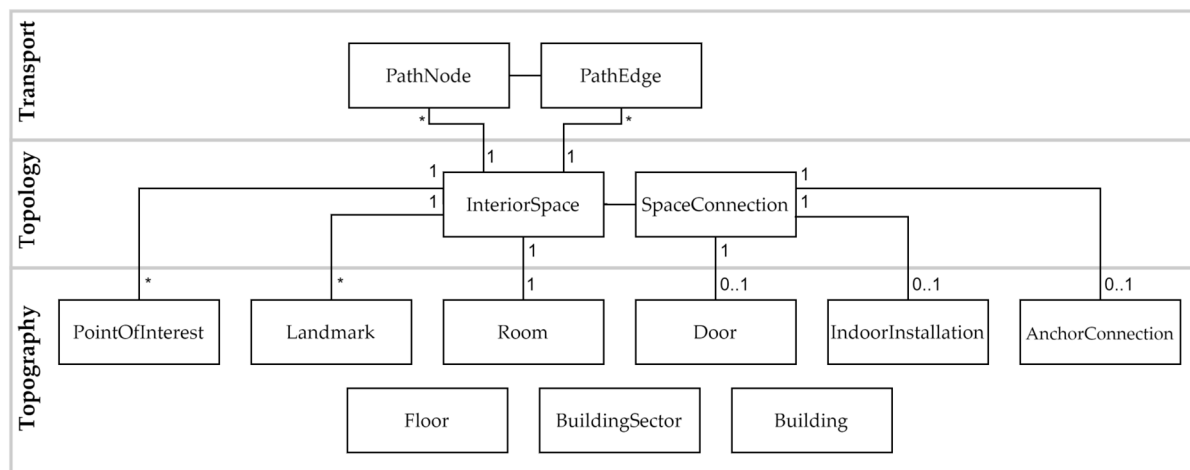


Figure 2. Class diagram for spatial data model used in the LIFT project (only the key classes and important associations are presented). Numbers define a multiplicity of classes in associations; '*' is equivalent to '0..*' and means 'zero or more instances'.

The graph representing the path network in the building consists of objects of *Path Node* and *Path Edge* classes. Geometries corresponding to *Path Edges* represent users' movement path fragments. They are modelled in accordance with natural pedestrian movement flow and should have geometries that support the creation of unequivocal navigational directions. *Path Nodes* model the connectivity of *Path Edges* and represent nodes in the graph. The principles of modelling the indoor path network and creating geometries to obtain the natural trajectory of the route were studied in [51].

The building topology graph is modelled with the *Interior Space* and *Space Connection* classes. In regular buildings, *Interior Space* represents rooms, corridors, or other well-defined parts of the building. At metro stations, most *Interior Spaces* are logical parts of the floor that are not necessarily separated by walls. However, some spaces can also be marked out by visible borders like turnstiles or dissimilar characteristics like differences between platforms turning into corridors. *Interior Spaces* in metro stations usually do not have names like rooms in regular buildings, but there are other features like ticket zones, which are restricted to ticket holders. Passages between *Interior Spaces* are represented by *Space Connections*. These can represent logical connections between logical spaces as well as physical connections like doors, gates, or turnstiles. Connections are also located between *Interior Spaces* on different levels or between exits from stations.

Attributes of *Space Connections* model the rules of traffic flow. Some connections have a limited time of crossing—doors and turnstiles are related to the metro operating times.

Others preclude some groups of people with special needs—small steps or turnstiles are a real barrier for people in wheelchairs. Based on the type, if necessary, some costs can be added to avoid situations when the route with the shortest distance crosses the ticket zone.

The classes from the topology layer of the model provide information on the location of other spatial objects. Associations with *Path Nodes* and *Path Edges* divide path networks by logical sub-spaces. This allows the algorithm to detect change in a room and supports the generation of a navigational direction. Change in a room is the most important direction for a user who expects the system to provide information about events such as passing through a door/gate or moving to another space with different characteristics. The topography layer classes can enrich guidance with spatial information about a room or a door or provide complex information involving descriptions of landmarks or POIs and spatial relations. Although the system does not implement a hierarchical landmark system, simple information about the presence of landmarks is included in the indoor guidance.

The next two classes, *Indoor Installation* and *Anchor Connection*, are also important. The *Indoor Installation* class represents connections between levels, such as stairs, elevators, escalators, or ramps. Such objects are modelled with three nodes—the entrance, the middle, and the exit. This type of modelling enables the connection of many floors and maintains the direction of the installation. The *Anchor Connection* class represents exits from the building. For metro stations, it represents exits to the ground level and connections to external transport systems, but also points on platforms that allow users to board the train and leave the station. The specific role of metro stations determines the fact that the usual route calculated by a user starts and ends in *Anchor Connections*.

The topography layer also consists of classes defining the building construction, such as *Building*, *Building Sector*, and *Floor*. They help to organise and validate the data and provide generic information for high-level descriptions of the calculated route. Figure 3 presents spatial data on the levels of the model and illustrates the information conveyed by classes within these levels.

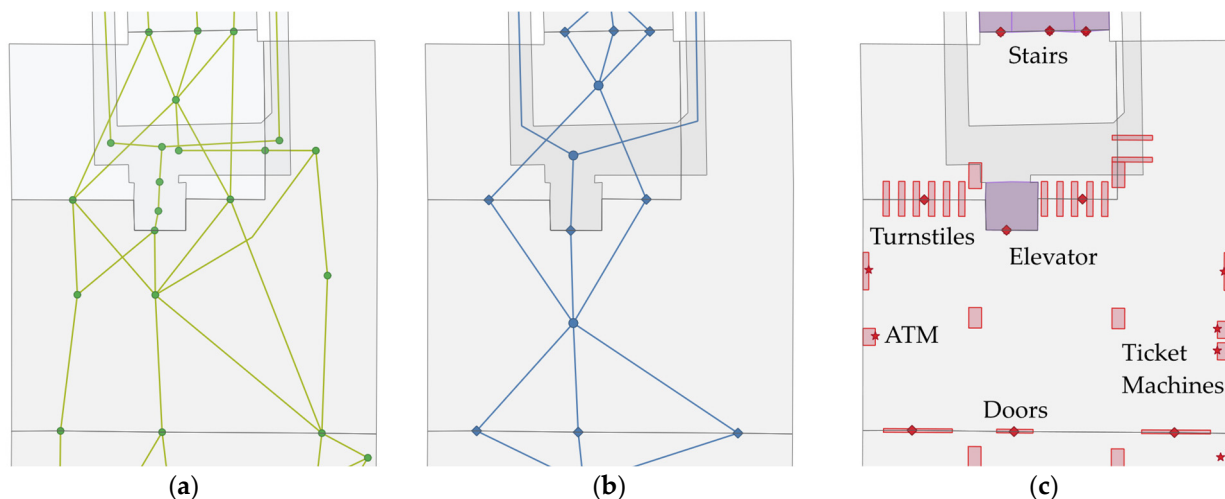


Figure 3. Data layers used in the system: (a) transport network—*Path Node* as a green circle and *Path Edge* as a green line; (b) room topology—*Interior Space* as a blue circle and *Space Connection* as a blue diamond; (c) building topography—*Indoor Installation* as a purple rectangle, *Furniture* as a red rectangle, *Point of Interest* and *Landmark* as red stars, and *Space Connection* as a red diamond.

3.8. A Framework of the Algorithm for Automatic Generation of Navigation Directions

The process of generating navigational directions has been divided into five stages, making the whole process transparent as well as easy to design, analyse, and maintain. Such an approach eases modifications of each stage independently after analysis of the outcomes of the tests following initial implementation; see Figure 4.

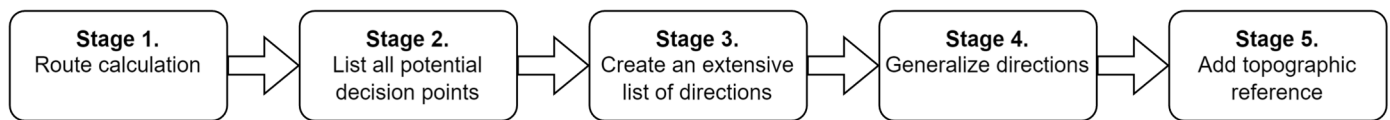


Figure 4. Stages of the algorithm for automatic generation of navigational directions.

The algorithm starts with route calculation (stage 1) and identification of all potential decision points (stage 2), which, in most cases, involves listing all nodes in the transport graph belonging to the route. The generation of navigation guidance is divided into two steps—the creation of an extensive list of directions for all decision points (stage 3) and the generalisation of the list (stage 4) to obtain the description of the route. In this stage, some directions are joined, useless directions are removed, and some other directions are left unchanged. The final step of the algorithm is to add topographic information to the direction (stage 5) to provide a comprehensive description of manoeuvres.

Stage 3 and stage 4 are implemented using a well-known design pattern, ‘chain of responsibility’. The algorithm is represented by a set of independent rules. In stage 3, for each decision point, the framework checks whether each rule can be applied and whether the application of a particular rule should complete the direction generation process for the decision point; see Figure 5. With this approach, different rules can be applied independently to create various types of directions and in addition, a given direction can be built following several rules that supplement it with new elements, e.g., making it more specific. An analogous rule-based process is used in stage 4. The consecutive rules try to apply different generalisations to the output of stage 3.

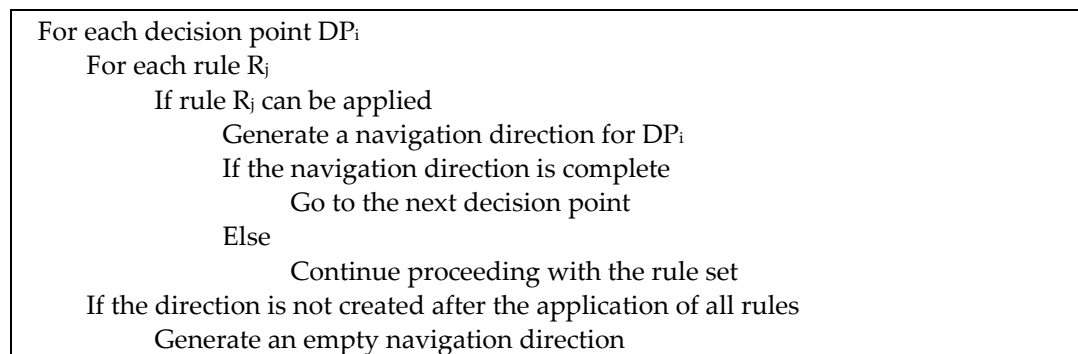


Figure 5. Rule-based algorithm—implementation of the pattern ‘chain of responsibility’.

3.9. Route Calculation Algorithm

The route is calculated with the Dijkstra algorithm finding the shortest path in the graph of transport network. In car navigation systems, more robust algorithms are used; for example, the A* algorithm or further optimisations that usually visit fewer graph nodes than the Dijkstra algorithm [60]. In the project, the performance of the Dijkstra algorithm is enough because navigation graphs for metro stations do not exceed 300 nodes.

The costs for edges are modelled based on the length of the edge and a factor dependent on the type of *Interior Space* or type of user. For example, it is recommended to go through the platform rather than the mezzanine if the route length is equal. There are different costs for *Indoor Installation*—usually, the escalator is faster than the stairs and the stairs are faster than the elevator. It is expected that routes should avoid excessive curves on the path and avoid passing through unnecessary objects like turnstiles, so the shape of path segments impacts the costs for the route calculation algorithm.

3.10. Considerations for Implementations of Rules

Implementation of the algorithm relies on creating the proper set of rules. When designing them, it is important to consider the following points discussed in previous sections:

- Navigation guidance provided for indoor environments, particularly metro stations, must take into account the specific conditions of confined spaces;
- The system creates navigation guidance to be consulted by users manually, not to be played automatically by a system based on a position obtained from an indoor positioning system;
- Metro stations are not the destination points of the route. Users go through stations to find a platform and board a train in a particular direction or exit the station.
- The navigation guidance should be usable for people with disabilities. The algorithm proposes a route considering the type of disability; then, the way the route directions are described should also take this disability into account;
- Turn-by-turn directions are not optimal. However, in some situations, they are acceptable;
- The full list of directions should be available for users to study for more time if needed;
- All directions should have spatial references to provide the option to see decision points on the map. They should also have pictures that allow users to find decision points in the neighbourhood.

For subsequent combination and generalisation of directions in stage 4, the algorithm uses rules of varying granularity. After testing and final adjustments, the number of rules in stage 3 of the algorithm was about 40, and in stage 4, about 30. It is expected that once the application is made available to users, the rules will be modified based on users' feedback.

The aim of this article is not to analyse the completeness of the rule set but to present some issues that are of interest in relation to the operation of the system. For this reason, only selected rules and their impact on the navigation guidance are presented in this article, while simple or obvious rules are omitted; see Table 2.

Table 2. Selected rules of the algorithm for automatic generation of navigational directions; rules 3-1 to 3-4 are used in stage 3, and rules 4-1 to 4-5 in stage 4.

Rule Id	Rule	Description
3-1	Change in the floor	Add a direction if floor change is detected. Provide detailed information using attributes of <i>Indoor Installation</i> .
3-2	Change in the room (logical space)	Add a direction based on <i>Space Connection</i> appearance. The direction is also added for two different consecutive <i>Interior Spaces</i> .
3-3	Go straight	If an angle between two consecutive <i>Path Edges</i> in route trajectory is close to 180 degrees, generate the 'Go straight' direction. Based on the tests, a tolerance of 30 degrees calculated at 15 m from the decision point was determined.
3-4	Turn left/right	If an angle between two consecutive <i>Path Edges</i> in route trajectory is close to more than 50 degrees, generate the 'Turn left/right' direction.
4-1	Combine multiple floor changes	If the route goes through many levels by one <i>Indoor Installation</i> combine it as a single floor change.
4-2	Remove useless 'Go straight'	Remove consecutive 'Go straight' directions unless some of them are created for decision points marked as 'Need direction here', which indicates the important point that requires a navigation direction.
4-3	Change two turns into one turnaround	Two consecutive turns are changed into one turnaround. Additional conditions must be fulfilled; for instance, the distance between turns does not exceed 10 m.
4-4	Replace 'turn left/right' with landmark-based direction	Each 'Turn left/right' direction is attempted to be changed. The algorithm verifies if (1) the orientation of a user body is defined—the preceding edges led through a corridor, etc.; (2) a landmark is available for the decision point.
4-5	Delete directions with repeating landmark	Remove consecutive directions based on the same landmark.

3.11. Implementation of Selected Rules for Generating Navigation Directions

In the previous section, selected rules are presented in the context of the creation of an extensive list of directions and the generalisation of this list. Two rules from each group were chosen to provide detailed implementation.

3.11.1. Implementation of Rule 3-1—Change in the Floor

Every change in the level triggers a direction providing detailed information related to this event. The input for this rule is two consecutive nodes with the attributes as follows: id as an ordinal number; vertical order as the order of levels in the building; indoor installation type indicating whether a transition is an escalator, elevator, stairs, or ramp; and floor needed for the name in the real world, displayed to the user. Finally, the outcome contains the movement direction (up or down), the type of *Indoor Installation*, and the name of the target floor. The pseudocode for this rule is provided in Figure 6.

```

If DP0.verticalorder != DP1.verticalorder
  If DP0.verticalorder - DP1.verticalorder < 0:
    Movement_direction = 'up'
  Elif DP0.verticalorder - DP1.verticalorder > 0:
    Movement_direction = 'down'

  If DP0 has attribute indoorinstallation:
    Type_of_indoor_installation = DP0.indoorinstallation

  If DP0.floorname != DP1.floorname
    Target_floor = DP1.floorname
  Else
    Target_floor = 'no floor name change'

```

Figure 6. Implementation of rule 3-1, change in the floor.

The last condition attracts attention. There may be a situation where there are clear visible stairs that connect levels with the same name. In such a case, the direction should be generated. Information about changing a floor to the same one may be misleading, so it is worth mentioning the type and direction of movement while omitting information about the name of the floor.

3.11.2. Implementation of Rule 3-4—Turn Left/Right

Rule 3-4 determines the angle between two consecutive *Path Edges* based on the geometry. For calculations, three consecutive nodes are used with the id as an ordinal number, point geometry, and number of neighbours expressed in an integer. The output is an angle and, consequently, the left/right direction. The pseudocode for this rule is provided in Figure 7.

```

If DP0.numberofneighbours > 2
  angle = get_angle (DP-1.geom, DP0.geom, DP1.geom)
  If angle > 50
    movement angle = 'right'
  Elif angle < -50
    movement angle = 'left'

```

Figure 7. Implementation of Rule 3-4, turn left/right.

The condition with the number of neighbours was introduced to avoid directions where the corridor naturally turns, and there are no alternative paths. The argument for

this approach is to reduce the final number of directions and increase the readability of the entire route.

3.11.3. Implementation of Rule 4-2—Remove Useless ‘Go Straight’

The main goal of this rule is to reduce repeating directions. Directions from two consecutive decision points are compared and finally, some directions are filtered out. The pseudocode for this rule is provided in Figure 8.

```

If DP0.directions has ‘go straight’ and DP1.directions has ‘go straight’
  If not (DP1.directions has ‘left’ or DP1.directions has ‘right’):
    delete ‘go straight’ from DP1.directions
  
```

Figure 8. Implementation of Rule 4-2, remove useless ‘go straight’.

In the metro station, long and straight corridors are common, so the rule eliminates multiple redundant directions that are not separated by a *Space Connection* like a door or stairs or are not detached by turns. This organises and increases the readability of the entire guidance.

3.11.4. Implementation of Rule 4-4—Replace ‘Turn Left/Right’ with a Landmark-Based Direction

This rule attempts to replace a ‘Turn left/right’ direction with a landmark or supplement it with additional information. The rule analyses three nodes: current, previous, and following. The outcome is direction enriched with relation to some object. The pseudocode for this rule is provided in Figure 9.

```

If DP0.id > 0
  If DP0.directions has ‘left/right’
    DP-1 has spaceconnection
    DP0.directions add ‘after -spaceconnection-’
If DP0.id < len (DP)–1
  If DP0.directions has ‘left/right’
    DP1 has spaceconnection
    DP0.directions add ‘before -spaceconnection-’
  
```

Figure 9. Implementation of Rule 4-4, replace ‘turn left/right’ with a landmark-based direction.

This generalisation handles adding relationships to relative turn information in most cases. The presented approach prefers the first landmark. Hence, in the given sequence: elevator, turn left, and door, the result will be ‘after exiting the elevator, turn left’ and ‘go through the door’ instead of ‘exit the elevator’ and ‘pass through the door on your left’.

4. Results

In this chapter, four routes are presented to demonstrate the operation of the algorithm in real-life cases. The first example, a typical route for a user in a wheelchair, allows us to trace the mechanism of the algorithm. The full route is presented to analyse the output of stage 3 and stage 4. The subsequent routes show typical problems that arose during rule design. In these cases, the descriptions focus on parts of the routes and selected directions only. For two cases, a universal solution was found, while the final problem required a compromise, resulting in the proposed solution not always producing optimal results.

4.1. Route 1—A Typical Route for a User in a Wheelchair

Route 1 is a common scenario for creating navigation guidance for a wheelchair passenger. The route starts at ground level, goes through a mezzanine, and ends on the platform and entrance to a metro train. The challenge of this route is the lack of an elevator

connecting the ground floor with the platform, and a user must change elevators on the interim floor. This requires crossing the entire floor and finding an elevator that goes to the platform (Figure 10).

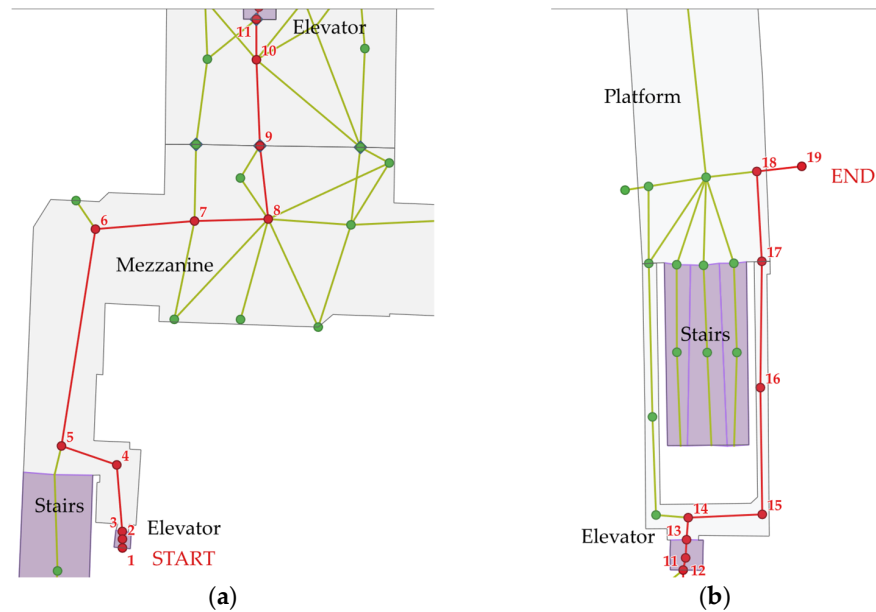


Figure 10. An example of a route for a wheelchair user that goes through (a) floor –1 (interim) and (b) floor –2 (platform); the maps present *Interior Spaces* (grey), *Indoor Installations* (purple), *Path Edges* (green and red for parts of the route), and *Path Nodes* (circles); decision points are labelled and marked in red.

Table 3 shows the navigation directions generated for the route. The route consists of 19 *Path Nodes* that produce 20 interim directions in stage 3. In decision point 6, the algorithm creates two directions: ‘turn right’ followed by ‘go straight’. At this stage, it makes sense to keep both directions due to the considerable long straight line to be passed after the turn. These directions are generalised in stage 4, and the final number of directions is reduced to 8. It is worth noting that in decision points 6 and 15, turns have been omitted due to the lack of *Path Edge* crossing. Thus, the movement to another direction is expected, and the direction is not needed. In decision point 8, the ‘turn left’ direction before a landmark is not needed. In decision point 8, the ‘turn left’ direction before a landmark is reduced to information that supplements the final direction: ‘on the left-hand side’.

Table 3. The output of the algorithm for Route 1.

Decision Point	Interim Directions (Output of Stage 3)	Final Directions After Generalisation in Stage 4
1	Change in the floor [down]	Go down with the elevator to the –1 level
2	Change in the floor [elevator]	
3	Change in the floor [–1]	
4	Turn left	-
5	Turn right	After exiting the elevator turn right
5	Go straight	
6	Turn right	Go straight and turn right
6	Go straight	
7	Go straight	Go straight and pass through the doors on the left-hand side
8	Turn left	
9	Pass the door	

Table 3. Cont.

Decision Point	Interim Directions (Output of Stage 3)	Final Directions After Generalisation in Stage 4
11	Change in the floor [down]	Go down with the elevator to the −2 level
12	Change in the floor [elevator]	
13	Change in the floor [−2]	
14	Turn right	After exiting the elevator turn right
15	Turn left	-
15	Go straight	Go straight into the platform
17	Change in the room [platform]	
18	Turn right	Get into the last car of a train on the right-hand side (direction: ‘Młociny’)
19	Go into train [last car]	

4.2. Route 2—‘Turn Left/Right’ Direction

As discussed earlier, navigation directions such as ‘turn left/right’ are generally not optimal as the interpretation of the heading can be ambiguous. However, in some cases, this type of direction can be helpful in finding the correct way. An example of such a situation is presented in Figure 11. A user approaching the ticket zone in the metro station notes three passages—the gate on the left for wheelchairs and two turnstiles—one in front of them and one more on the right. Moreover, there is no landmark that can be used for indication of the passage to be taken. The only solution is to base the direction on a ‘turn left/right’ message.



Figure 11. Entrance to the ticket zone in the Politechnika metro station.

In order to solve this issue, it was required to identify conditions that trigger a specific rule. In this case, the route leads through an *Interior Space* to a *Space Connection* of type ‘ticket zone’, and there are more *Space Connections* of the same type nearby. The algorithm should attempt to distinguish them, preferably by indicating a landmark, and if there is no landmark, a ‘turn left/right’ direction should be provided.

The other problem observed while testing the implementation of such a rule was the accurate determination of turns when the geometry of the path network was analysed. While the geometric properties of *Path Edges* typically suffice to determine the angle of a turn, special consideration is needed when aligning with turnstiles or doors. This is because spatial connections for stairs or elevators tend to be more linear, whereas doors and turnstiles are represented as points. In Figure 12, at a point located at wide turnstiles on the left, the angle of the consecutive *Path Edges* is less than 30 degrees. However, the expected

direction after going through the turnstiles is ‘turn right’. This problem was solved so that the *Indoor Installation* stores information about its azimuth and, on this basis, an additional turn direction can be created.

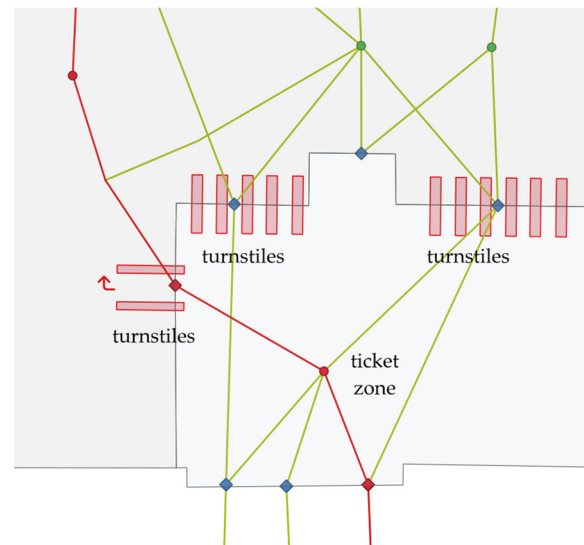


Figure 12. Map with the path network for the ticket zone of the Politechnika metro station; the map presents *Interior Spaces* (grey), *Path Edges* (green and red for parts of the route), and *Path Nodes* (circle); the red arrow indicates the decision point where the ‘turn right’ direction should be added.

4.3. Route 3—Moving to Another Floor

When a route requires changing floor, a user expects a single direction indicating the final floor for the manoeuvre, for example, ‘Go down with the elevator/stairs to the –2 level’. If the route requires a change of more than one floor, the directions in stage 3 of the algorithm are generated for each floor separately. There are a few cases to consider:

- **Elevators:** These are represented on each floor separately and connected in one graph. A route that uses an elevator to traverse a few floors contains a series of *Path Nodes* for each floor. Such a sequence is easy to detect and is replaced with a single navigation direction;
- **Well-defined staircase with stairs connecting consecutive floors:** There are a few possible configurations of stairs. In Figure 13a, the stairs reach the next floor (or mezzanine) after two turns, but all these manoeuvres are irrelevant from a navigational point of view. The algorithm generalises the navigational direction using the attributes of the *Interior Space*, indicating whether they are parts of the staircase and analysing distances between consecutive stairs;
- **A different situation is presented in Figure 13b.** The possible connections between the ground floor and floor –2 fork to opposite sides of the stairs. When the user goes down, there is no alternative path, so directions should be generalised. However, in the case of going up, it is important to take the correct exit from the station. Hence, the information on whether to take a left or right turn is essential;
- **Subsequent stairs:** Although some stairs are connected to a single navigational direction, in some cases, the algorithm should consider the consecutive stairs to be separate objects, which is an expected outcome when they are of different types (stairs, escalators, moving walkways) or if the distance between them is big enough. An example of such a situation is shown in Figure 14. The tests led to the decision to implement the rules to generalise the directions if the distance is less than 12 m, regardless of whether they are of the same type or not.

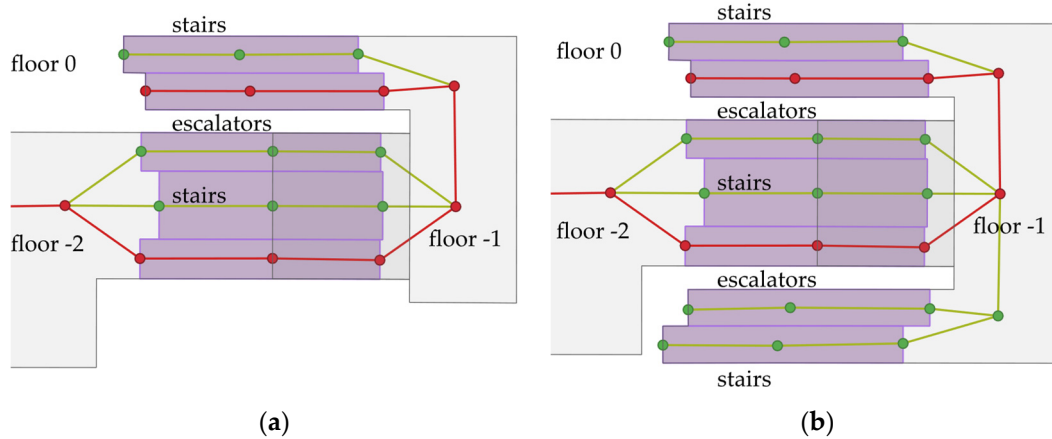


Figure 13. Examples of floor connections: (a) staircase with separated escalators, (b) staircase with subsequent escalators; the maps present *Interior Spaces* (grey), *Indoor Installations* (purple), *Path Edges* (green and red for parts of the route), and *Path Nodes* (circles); decision points are marked in red.

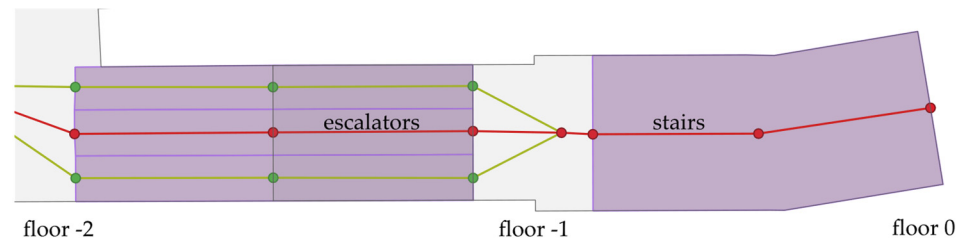


Figure 14. Examples of floor connections: subsequent stairs of a different type; the maps present *Interior Spaces* (grey), *Indoor Installations* (purple), *Path Edges* (green and red for parts of the route), and *Path Nodes* (circles); decision points are marked in red.

4.4. Route 4—The First Direction After Exiting the Train

A typical use case of navigation in a metro station is exiting the train on the platform and finding a way out on the ground level. Empirical tests have shown that the critical moment of the process is the first decision of users. They leave a train and, to perform the first manoeuvre correctly, must orient themselves in a space that is new to them. The problem is even more complicated since the users' location on the platform is not known. A user can enter the train at the beginning, middle, or end and then move to a different part of the train during the journey. This causes the user to start navigating from an unknown position on the platform.

The problem becomes even more complex if we consider the different configurations of stairs to exit the platform. If the stairs are located at the end of the platform, as in Figure 15, no matter where the passenger gets off the train, they are supposed to turn right. If, however, the stairs are located on the platform between the platform ends, as in Figure 16, then the user should turn right or left depending on where they get off the train. In addition, if the user gets off at the end of the train, the indication to go left towards the stairs is ambiguous because they pass other stairs beforehand.

The adopted solution to the problem was based on moving away from specifying the direction in which the passenger should turn after exiting the train. Instead, the direction contains an absolute indication of the position of the stairs on the platform relative to the train. The tests showed that it was sufficient to describe the position of the stairs with one of five terms: 'the start of the platform', 'the second carriage', 'the middle of the platform', 'the penultimate carriage', and 'the end of the platform'. Thus, the full guideline is as follows: 'proceed to the stairs at the second carriage' or 'proceed to the stairs at the end of the platform'. It is worth noting that, where possible, it is useful to refer to the platform, not the train, as it may have left in the meantime. During the tests, users intuitively understood

that the start and end of the platform referred to the direction of the train. The tests also showed that users were able to identify the position of the second and penultimate carriage at the station without much effort, whereas references to other carriages required counting and discouraged users from trying to understand the message. Hence, the authors decided to use a reference to one of the five staircase positions.

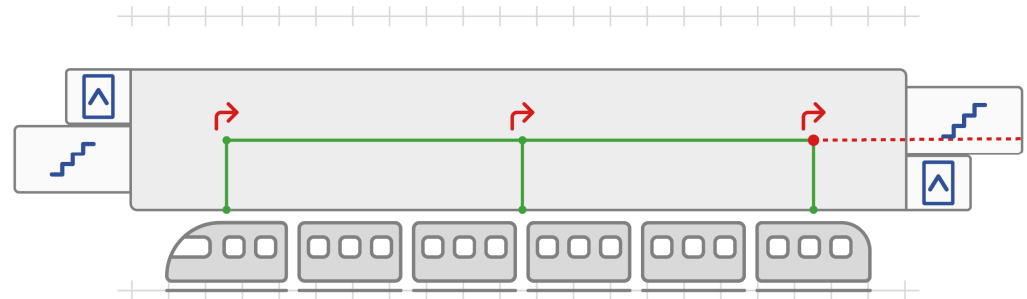


Figure 15. A metro station with stairs at a platform end—a passenger should go right to the stairs regardless of their position after leaving a train; the map presents *Path Edges* (green), and *Path Nodes* (circles); red arrows indicate decision points where the ‘turn right’ directions should be generated depending on the route course; the red dotted line represents the exit *Path Edge* common for all routes.

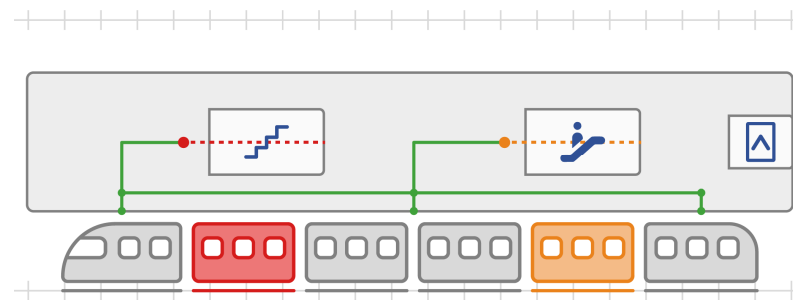


Figure 16. A metro station with stairs between platform ends—the direction of the first turn depends on the passenger’s initial position after leaving a train; the map presents *Path Edges* (green) and *Path Nodes* (circles); the red dotted line represents the exit *Path Edge* for routes for passengers travelling near the red carriage, the orange dotted line – near the orange carriage.

As the first navigation direction is independent of the user’s starting position, the route is not drawn from the train exit but from the stairs. In Figures 11 and 12, the start of the route is marked with a red dot.

The second problem related to the start of the route on the platform is the choice of the appropriate stairs. If there are several stairs on the platform leading to different parts of the station, then the situation is simplified, as it is optimal to choose particular stairs in order to exit the station at the right point on the ground level. However, there are stations where the choice of stairs on the platform is irrelevant, and passengers in this situation choose the stairs closest to them and would expect the algorithm to work the same. Since it is not known from which part of the train the user has disembarked, without a precise positioning system it is not possible to solve this problem perfectly. The system, however, estimates the user’s position at the station considering the navigation guidance at the station where the metro journey commenced.

The proposed solutions to both problems do not always work. Determining the platform position in relation to the train depends on the length of the carriages and can be cumbersome once the train has departed. In contrast, the estimation of the user’s position assumes that they do not walk too far on the starting platform or within the train. Despite the significant drawbacks of the approach taken, it has improved navigation performance and users’ understanding of navigational directions.

5. Discussion

Urban commuter planners provide comprehensive information on all modes of transport in a city, but they are usually limited to routing functions. At the same time, an increasing number of public buildings are providing indoor navigation systems. Such applications use positioning engines, which, despite research over the past decade, still provide poor positioning accuracy when run on smartphones—widely available devices. A large proportion of indoor navigation systems use turn-by-turn guidance due to the much higher cost of basing navigation guidance on landmarks. This approach, together with a positioning system with poor accuracy, limits the ability to popularise indoor navigation systems in public buildings.

The application, which is the result of the research, focuses on pedestrian navigation in metro stations and takes into account the expectations of people with special needs. By design, the application will be used without a positioning engine, and users will determine their location based on the map and landmarks, aided by photographs attached to the route decision points. The authors focused on implementing algorithms to generate navigational directions automatically, and through the support of people with disabilities, it was possible to create an application that works acceptably for the target group. It is still necessary to collect and maintain the spatial database of landmarks, in addition to taking photos and updating them periodically. However, the idea is that automated algorithms help to reduce this cost, as a limited number of landmarks are needed to generate guidance.

A natural direction for further development of the system is integration with a positioning system. The lack of presentation of the user's position in the current application can be seen as a drawback because users accustomed to other systems may cease trying to use this application because of this missing function. On the other hand, current positioning algorithms using Wi-Fi networks or Bluetooth beacons offer position detection with low accuracy. A system generating navigation guidance should take into account the accuracy and the characteristics of a calculated position. Hence, integration of the positioning system with the navigation system is not a straightforward task and requires additional research and testing. Navigation guidance must be built in such a way that potential position error does not prevent its interpretation. The system was built with the assumption that the user recognises his surroundings based on the description of the direction and the photo. This should help in the potential integration of the solution; however, the system has not been tested with the positioning engine, which makes it difficult to estimate the work needed.

During the development of the algorithm, various rule modifications were considered, and their impact on the final indoor guidance was analysed. In some cases, satisfactory solutions have been developed; in others, specific approaches have been adopted with the expectation that end users will assess their usefulness. When the system is made available, a number of comments from users are expected. The feedback will be analysed and influence the modification of the rule set.

Each time a developer decides to provide a change in the spatial data or change in the algorithm's rules, they must consider side effects—how the change impacts other routes. The authors came to the conclusion that predefined navigational directions can improve the system maintenance process. This feature allows one to define a direction explicitly and link it to a decision point. Whenever the algorithm encounters a predefined direction, it breaks the rule-based inference and adopts the direction designed by the operator.

This system was intended to be built for stations of the Warsaw Metro, and the authors were focused on creating the best possible solution operating within this scope. This limits the solution to metro stations of a specific architectural construction style and limited complexity. However, the authors will endeavour to obtain further funding for research into navigation at other rail transport stations. Navigation at railway stations and terminals with complex structures—multi-storey and connecting multiple railways, subways, and other modes of transport—is a planned direction of further development.

6. Conclusions

This study acknowledges the unique challenges posed by metro stations, such as their complex architecture, lack of natural light, and limited visibility. These factors make traditional navigation methods less effective, thereby highlighting the importance of innovative solutions like the landmark-based approach. This research also points out the need for inclusivity in navigation systems, ensuring that they cater to the diverse needs of all users, including those with disabilities.

The analysis of the performance of the implemented algorithm for different cases leads to the conclusion that attention should be paid not only to the construction of the rules that create the navigational directions but also to the data model. Establishing correct relationships between the layers of the data model, setting appropriate attribute values, and proper preparation of the geometry enable the optimal operation of the algorithm. This relationship is evident in the examples discussed. Correctly modelled paths make it easier to determine whether there is a left turn, a right turn, or a straight run. Such a determination may depend on the subjective feeling of the user at a given point indoors, so it is up to the cartographer to prepare the data model to anticipate the user's expectations and delineate the geometry accordingly. Similarly, when generalising staircases, it is possible to force the algorithm to work in a specific way by setting appropriate attributes for topographic objects.

The algorithm presented in this paper has been implemented and tested on mobile devices for different routes at different metro stations in Warsaw. After several testing iterations, the project team developed a quality algorithm, allowing the mobile application to be made available on Android Play and the AppStore. The system launch is planned for the end of 2024. Making the application available to customers will enable the collection of broader feedback necessary for maintaining and further developing the technology. The unique experience gained during this research will be used to design solutions for other cities with more extensive underground urban transport infrastructure.

Author Contributions: Conceptualisation, J.B.M.; methodology, J.B.M.; software, B.W.; validation, B.W.; investigation, J.B.M. and B.W.; resources, B.W.; data curation, B.W.; writing—original draft preparation, J.B.M. and B.W.; writing—review and editing, J.B.M. and B.W.; visualisation, B.W.; supervision, J.B.M.; project administration, B.W.; funding acquisition, B.W. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by The National Center for Research and Development (NCBR), grant number “Rzeczy są dla ludzi/0049/2020”, and the APC was funded by NCBR.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The raw data supporting the conclusions of this article will be made available by the authors on request.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Klippel, A. Wayfinding Choremes. In *Spatial Information Theory. Foundations of Geographic Information Science*; Kuhn, W., Worboys, M.F., Timpf, S., Eds.; Springer: Berlin/Heidelberg, Germany, 2003; pp. 301–315.
2. Fang, Z.; Li, Q.; Zhang, X. A Multiobjective Model for Generating Optimal Landmark Sequences in Pedestrian Navigation Applications. *Int. J. Geogr. Inf. Sci.* **2011**, *25*, 785–805. [\[CrossRef\]](#)
3. Gu, F.; Kealy, A.; Khoshelham, K.; Shang, J. Efficient and Accurate Indoor Localization Using Landmark Graphs. *Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci.* **2016**, *XLI-B2*, 509–514. [\[CrossRef\]](#)
4. Radoczy, V. How to Design a Pedestrian Navigation System for Indoor and Outdoor Environments. In *Location Based Services and TeleCartography*; Gartner, G., Cartwright, W., Peterson, M.P., Eds.; Lecture Notes in Geoinformation and Cartography; Springer: Berlin/Heidelberg, Germany, 2007; pp. 301–316; ISBN 978-3-540-36727-7.
5. Russo, D.; Zlatanova, S.; Clementini, E. Route Directions Generation Using Visible Landmarks. In Proceedings of the Sixth ACM SIGSPATIAL International Workshop on Indoor Spatial Awareness; Association for Computing Machinery, New York, NY, USA, 4 November 2014; pp. 1–8.

6. Dudas, P.M.; Mahsa, G.; Hassan, A.K. ONALIN: Ontology and algorithm for indoor routing. In Proceedings of the 2009 Tenth International Conference on Mobile Data Management: Systems, Services and Middleware, Taipei, Taiwan, 18–20 May 2009; pp. 720–725.
7. Yang, L.; Worboys, M. A Navigation Ontology for Outdoor-Indoor Space: (Work-in-Progress). In Proceedings of the 3rd ACM SIGSPATIAL International Workshop on Indoor Spatial Awareness; Association for Computing Machinery, New York, NY, USA, 1 November 2011; pp. 31–34.
8. Viaene, P. Indoor Landmarks: The Study and Implementation of Wayfinding Aids. Ph.D. Thesis, Ghent University, Ghent, Belgium, 2018.
9. Dubey, R.K.; Sohn, S.S.; Thrash, T.; Hoelscher, C.; Kapadia, M. Identifying Indoor Navigation Landmarks Using a Hierarchical Multi-Criteria Decision Framework. In Proceedings of the 12th ACM SIGGRAPH Conference on Motion, Interaction and Games; Association for Computing Machinery, New York, NY, USA, 28 October 2019; pp. 1–11.
10. Staroverov, A.; Panov, A.I. Hierarchical Landmark Policy Optimization for Visual Indoor Navigation. *IEEE J. Mag. IEEE Access* **2022**, *10*, 70447–70455. [\[CrossRef\]](#)
11. Liu, B.; Ding, L.; Meng, L. Spatial Knowledge Acquisition with Virtual Semantic Landmarks in Mixed Reality-Based Indoor Navigation. *Cartogr. Geogr. Inf. Sci.* **2021**, *48*, 305–319. [\[CrossRef\]](#)
12. Gao, L.; Konomi, S. Mapless Indoor Navigation Based on Landmarks. In *Distributed, Ambient and Pervasive Interactions. Smart Living, Learning, Well-Being and Health, Art and Creativity*; Streitz, N.A., Konomi, S., Eds.; Springer International Publishing: Cham, Switzerland, 2022; pp. 53–68.
13. Serrão, M.; Rodrigues, J.M.F.; Rodrigues, J.I.; du Buf, J.M.H. Indoor Localization and Navigation for Blind Persons Using Visual Landmarks and a GIS. *Procedia Comput. Sci.* **2012**, *14*, 65–73. [\[CrossRef\]](#)
14. Dong, H.; Ganz, A. Automatic Generation of Indoor Navigation Instructions for Blind Users Using a User-Centric Graph. In Proceedings of the 2014 36th Annual International Conference of the IEEE Engineering in Medicine and Biology Society, Chicago, IL, USA, 26–30 August 2014; pp. 902–905.
15. Anup, S.; Goel, A.; Padmanabhan, S. Visual Positioning System for Automated Indoor/Outdoor Navigation. In Proceedings of the TENCON 2017–2017 IEEE Region 10 Conference, George Town, Malaysia, 5–8 November 2017; pp. 1027–1031.
16. CSUN Assistive Technology Conference, © 2024 California State University Northridge, 18111 Nordhoff Street Northridge. Available online: <https://www.csun.edu/cod> (accessed on 11 September 2024).
17. Bruglieri, M.; Colomi, A.; Lia, F.; Luè, A. A Multi-Objective Time-Dependent Route Planner: A Real World Application to Milano City. *Transp. Res. Procedia* **2014**, *3*, 460–469. [\[CrossRef\]](#)
18. Bucher, D.; Jonietz, D.; Raubal, M. A Heuristic for Multi-Modal Route Planning. In *Progress in Location-Based Services 2016*; Gartner, G., Huang, H., Eds.; Springer International Publishing: Cham, Switzerland, 2017; pp. 211–229.
19. Benchimol, P.; Amrani, A.; Khouadja, M. A Multi-Criteria Multi-Modal Predictive Trip Planner: Application on Paris Metropolitan Network. In Proceedings of the 2021 IEEE International Smart Cities Conference (ISC2), Manchester, UK, 7–10 September 2021; pp. 1–4.
20. On the Path to Develop a Micromobility Journey Planner for Madrid: A Tool to Estimate, Visualize, and Analyze Cycling and Other Shared Mobility Services' Flow | Journal of Transport and Land Use. Available online: <https://www.jtlu.org/index.php/jtlu/article/view/2451> (accessed on 11 September 2024).
21. Santos-González, I.; Rivero-García, A.; Caballero-gil, P. Secure Indoor Location for Airport Environments. In Proceedings of the 2018 4th International Conference on Big Data Innovations and Applications (Innovate-Data), Barcelona, Spain, 6–8 August 2018; pp. 60–64.
22. Molina, B.; Olivares, E.; Palau, C.E.; Esteve, M. A Multimodal Fingerprint-Based Indoor Positioning System for Airports. *IEEE Access* **2018**, *6*, 10092–10106. [\[CrossRef\]](#)
23. Law, I.; Loe, H. Generic Airport Location-Based Services Using an Integrated Indoor Positioning System. *J. Airt. Manag.* **2020**, *14*, 123–140. [\[CrossRef\]](#)
24. Chilakapati, A.K.; Iyer, V. Appreciation of Passenger's Intuitive Navigation In 'Ticketing Cum Entrance Lobby' Of Railway Station—A Case Study of Vijayawada Railway Junction, India—ProQuest. *NeuroQuantology* **2022**, *20*, 4526–4545. [\[CrossRef\]](#)
25. Wang, W.; Klettner, S.; Gartner, G.; Fian, T.; Hauger, G.; Angelini, A.E.N.; Söllner, M.; Florack, A.; Skok, M.; Past, M. Towards a User-Oriented Indoor Navigation System in Railway Stations. In Proceedings of the Adjunct Proceedings of the 15th International Conference on Location Based Services (LBS 2019), Wien, Austria, 11–13 November 2019; Gartner, G., Huang, H., Eds.; Vienna University of Technology: Vienna, Austria, 2019; pp. 63–70. [\[CrossRef\]](#)
26. Kacorri, H.; Ohn-Bar, E.; Kitani, K.M.; Asakawa, C. Environmental Factors in Indoor Navigation Based on Real-World Trajectories of Blind Users. In Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems, Montreal, QC, Canada, 21–26 April 2018; ACM: Montreal, QC, Canada, 2018; pp. 1–12.
27. Ahmed, S. Wayfinding and Indoor Navigation for Persons with Visual Impairments: An Overview of Support Innovations by Mada. *Nafath* **2023**, *8*. [\[CrossRef\]](#)
28. Yoon, J.; Kim, S. Practical and Accurate Indoor Localization System Using Deep Learning. *Sensors* **2022**, *22*, 6764. [\[CrossRef\]](#) [\[PubMed\]](#)
29. Koutris, A.; Siozos, T.; Kopsinis, Y.; Pikrakis, A.; Merk, T.; Mahlig, M.; Papaharalabos, S.; Karlsson, P. Deep Learning-Based Indoor Localization Using Multi-View BLE Signal. *Sensors* **2022**, *22*, 2759. [\[CrossRef\]](#) [\[PubMed\]](#)

30. Talla-Chumpitaz, R.; Castillo-Cara, M.; Orozco-Barbosa, L.; García-Castro, R. A Novel Deep Learning Approach Using Blurring Image Techniques for Bluetooth-Based Indoor Localisation. *Inf. Fusion* **2023**, *91*, 173–186. [\[CrossRef\]](#)
31. Kuriakose, B.; Shrestha, R.; Sandnes, F.E. DeepNAVI: A Deep Learning Based Smartphone Navigation Assistant for People with Visual Impairments. *Expert Syst. Appl.* **2023**, *212*, 118720. [\[CrossRef\]](#)
32. Said, Y.; Atri, M.; Albahar, M.A.; Ben Atitallah, A.; Alsariera, Y.A. Obstacle Detection System for Navigation Assistance of Visually Impaired People Based on Deep Learning Techniques. *Sensors* **2023**, *23*, 5262. [\[CrossRef\]](#)
33. Akilandeswari, J.; Jothi, G.; Naveenkumar, A.; Sabeenian, R.S.; Iyyanar, P.; Paramasivam, M.E. Design and Development of an Indoor Navigation System Using De-noising Autoencoder Based Convolutional Neural Network for Visually Impaired People. *Multimed. Tools Appl.* **2022**, *81*, 3483–3514. [\[CrossRef\]](#)
34. De Oliveira, L.C.; Andrade, A.O.; de Oliveira, E.C.; Soares, A.; Cardoso, A.; Lamounier, E. Indoor Navigation with Mobile Augmented Reality and Beacon Technology for Wheelchair Users. In Proceedings of the 2017 IEEE EMBS International Conference on Biomedical & Health Informatics (BHI), Orland, FL, USA, 16–19 February 2017; pp. 37–40.
35. Kim, E.Y. Wheelchair Navigation System for Disabled and Elderly People. *Sensors* **2016**, *16*, 1806. [\[CrossRef\]](#)
36. Şahin, H.İ.; Kavsaoglu, A.R. Autonomously Controlled Intelligent Wheelchair System for Indoor Areas. In Proceedings of the 2021 3rd International Congress on Human-Computer Interaction, Optimization and Robotic Applications (HORA), Ankara, Turkey, 11–13 June 2021; pp. 1–6.
37. Pacini, F.; Dini, P.; Fanucci, L. Design of an Assisted Driving System for Obstacle Avoidance Based on Reinforcement Learning Applied to Electrified Wheelchairs. *Electronics* **2024**, *13*, 1507. [\[CrossRef\]](#)
38. Sevastopoulos, C.; Acharya, S.; Makedon, F. An RGB-D Fusion System for Indoor Wheelchair Navigation. In Proceedings of the 16th International Conference on Pervasive Technologies Related to Assistive Environments, Corfu, Greece, 5–7 July 2023; Association for Computing Machinery: New York, NY, USA, 2023; pp. 187–191.
39. Mautz, R. *Indoor Positioning Technologies*; IEEE: New York, NY, USA, 2012.
40. Pasricha, S. Overview of Indoor Navigation Techniques. In *Position, Navigation, and Timing Technologies in the 21st Century*; John Wiley & Sons, Ltd.: Hoboken, NJ, USA, 2020; pp. 1141–1170; ISBN 978-1-119-45855-5.
41. Nagel, C.; Becker, T.; Kaden, R.; Li, K.-J.; Lee, J.; Kolbe, T.H. Requirements and Space-Event Modeling for Indoor Navigation. How to Simultaneously Address Route Planning, Multiple Localization Methods, Navigation Contexts, and Different Locomotion Types. 2010. Available online: <https://mediatum.ub.tum.de/doc/1155642/737344.pdf> (accessed on 1 October 2024).
42. Lee, J.; Becker, T.; Nagel, C.; Kolbe, T.H.; Sisi, Z.; Li, K.-J. OGC® IndoorGML; Version 1.0.; Open Geospatial Consortium: Wayland, MA, USA, 2014.
43. Gröger, G.; Kolbe, T.H.; Czerwinski, A.; Nagel, C. *OpenGIS City Geography Markup Language (CityGML) Encoding Standard*; Version 1.0.0; International OGC Standard; Open Geospatial Consortium, Doc: Wayland, MA, USA, 2008.
44. Kang, H.-K.; Li, K.-J. A Standard Indoor Spatial Data Model—OGC IndoorGML and Implementation Approaches. *ISPRS Int. J. Geo-Inf.* **2017**, *6*, 116. [\[CrossRef\]](#)
45. Liu, L.; Zlatanova, S.; Li, B.; van Oosterom, P.; Liu, H.; Barton, J. Indoor Routing on Logical Network Using Space Semantics. *ISPRS Int. J. Geo-Inf.* **2019**, *8*, 126. [\[CrossRef\]](#)
46. Yan, J.; Zlatanova, S.; Diakité, A. A Unified 3D Space-Based Navigation Model for Seamless Navigation in Indoor and Outdoor. *Int. J. Digit. Earth* **2021**, *14*, 985–1003. [\[CrossRef\]](#)
47. Swobodzinski, M.; Raubal, M. An Indoor Routing Algorithm for the Blind: Development and Comparison to a Routing Algorithm for the Sighted. *Int. J. Geogr. Inf. Sci.* **2009**, *23*, 1315–1343. [\[CrossRef\]](#)
48. Krüminaitė, M.; Zlatanova, S. Indoor Space Subdivision for Indoor Navigation. In Proceedings of the Sixth ACM SIGSPATIAL International Workshop on Indoor Spatial Awareness, Dallas-Fort Worth, TX, USA, 4 November 2014; Association for Computing Machinery: New York, NY, USA, 2014; pp. 25–31.
49. Goetz, M.; Zipf, A. Formal Definition of a User-Adaptive and Length-Optimal Routing Graph for Complex Indoor Environments. *Geo-Spat. Inf. Sci.* **2011**, *14*, 119–128. [\[CrossRef\]](#)
50. Yang, L.; Worboys, M. Generation of Navigation Graphs for Indoor Space. *Int. J. Geogr. Inf. Sci.* **2015**, *29*, 1737–1756. [\[CrossRef\]](#)
51. Dąbrowska, A. Model Grafu Nawigacyjnego Dla Celów Nawigacji Wewnątrz Gmachu Głównego Politechniki Warszawskiej, Praca Inżynierska. Bachelor’s Thesis, Technical University of Warsaw, Warszawa, Poland, 2020.
52. Anastassova, M.; Roselier, S. A Preliminary User Evaluation of Vibrational Patterns Transmitting Spatial Information. In Proceedings of the Workshop at MobileHCI 2010, Lisbon, Portugal, 7–10 September 2010; Citeseer: Berkeley, CA, USA, 2010; p. 29.
53. Pielot, M.; Boll, S. In Fifty Metres Turn Left”: Why Turn-by-Turn Instructions Fail Pedestrians. Haptic, Audio and Visual Interfaces for Maps and Location Based Services 2010. Available online: <https://uol.de/f/2/dept/informatik/ag/medien/download/pubs/Pielot2010-In-Fifty-Metres-Turn-Left.pdf> (accessed on 1 October 2024).
54. Frankenstein, J.; Brüssow, S.; Ruzzoli, F.; Hölscher, C. The Language of Landmarks: The Role of Background Knowledge in Indoor Wayfinding. *Cogn. Process* **2012**, *13*, 165–170. [\[CrossRef\]](#)
55. Wiener, J.M.; Hölscher, C.; Büchner, S.; Konieczny, L. Gaze Behaviour during Space Perception and Spatial Decision Making. *Psychol. Res.* **2012**, *76*, 713–729. [\[CrossRef\]](#)
56. Ohm, C.; Müller, M.; Ludwig, B.; Bienk, S. Where is the Landmark? Eye Tracking Studies in Large-Scale Indoor Environments. In Proceedings of the ET4S 2014, Vienna, Austria, 23 September 2014; Volume 1241, pp. 47–51.

57. Gotlib, D.; Gnat, M.; Marciniak, J. The Research on Cartographical Indoor Presentation and Indoor Route Modeling for Navigation Applications. In Proceedings of the 2012 International Conference on Indoor Positioning and Indoor Navigation (IPIN), Sydney, Australia, 13–15 November 2012; pp. 1–7.
58. Gotlib, D.; Gnat, M. Spatial Database Modeling for Indoor Navigation Systems. *Rep. Geod. Geoinformatics* **2013**, *95*, 49–63. [[CrossRef](#)]
59. Gotlib, D.; Gnat, M. Koncepcja Wielofunkcyjnego Systemu Informacji Przestrzennej Wspomagającego Zarządzanie i Użytkowanie Nieruchomości Politechniki Warszawskiej. *Rocz. Geomatyki-Ann. Geomat.* **2018**, *16*, 299–318.
60. Zeng, W.; Church, R.L. Finding Shortest Paths on Real Road Networks: The Case for A*. *Int. J. Geogr. Inf. Sci.* **2009**, *23*, 531–543. [[CrossRef](#)]

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.