TECHNISCHE UNIVERSITÄT WIEN
Vienna University of Technology

Diplomarbeit

Towards a Cognitively-Sound, Landmark-Based Indoor Navigation System

unter der Leitung von

Univ. Ass. Dipl.-Ing. Dr.-Ing. Paolo Fogliaroni

Fakultät für Mathematik und Geoinformation

von

Maja Pavlin

Wien, im December 2016 Unterschrift

DECLARATION OF AUTHORSHIP

I, **MAJA PAVLIN**, hereby declare that I am the author of the graduation thesis entitled: »*Towards a Cognitively-Sound, Landmark-Based Indoor Navigation System*«.

Vienna, December 2016

(signature)

BIBLIOGRAPHIC – DOCUMENTALISTIC INFORMATION AND ABSTRACT

Author:	**Maja Pavlin**
Supervisor:	**Univ.Ass. Dipl.-Ing. Dr.-Ing. Paolo Fogliaroni**
Title:	**Towards a Cognitively-Sound, Landmark-Based Indoor Navigation System**
Document type:	**Diploma Thesis**

Abstract

While outdoor navigation has received, by now, much attention and reached quite standardized technologies and techniques to automatic localization and navigation (mainly based on GPS-tracking), the problem of navigating people indoor has received relatively less attention. The very different nature of outdoor and indoor spaces makes the problem of navigating people through indoor environments also very different from the outdoor navigation problem. Most of the indoor navigation techniques developed so far try to replicate the satellite triangulation applicable outdoor by replacing GPS signal by other signals such as WIFI, RFID, and Bluetooth. Another type of approach tries, as soon as GPS coverage is lost, to build movement trajectories from sensors available in modern mobile devices such as compasses, magnetometers, and accelerometers.

This work explores and analyses a different solution to navigating people in indoor environments, one that does not require installation and usage of any extra devices in the environments. The idea is to navigate people based on qualitative coordinates, which can be described as what one can see at a certain position. The main requirement to exploit this technique is to have a detailed database of the environment in order to compute visibility between objects. With that we divide space into regions, i.e. extract qualitative coordinates. By using these regions, it is possible to create landmark-based instructions, which are, according to the literature, cognitively more appropriate (in the sense that they produce a lower cognitive load on the user) instructions than metric instructions.

Since visibility is a spatial property directly related to 3D space, we present the idea of visibility-based navigation by using the data in 3D. More specifically, we analyse the extension to 3D space of an existing 2D qualitative localization and navigation system based on visibility information, we present the main challenges that need to be tackled in order to get a functional implementation of such a landmark-based navigation system based on 3D visibility information, and address a subset of them.

The analysis shows we do not need to compute visibility between all objects in the database, but only between landmarks and obstacles. Therefore, we propose the usage of some criteria to minimize produced visibility regions. Objects with specific role in indoor environment should be used to simplify some regions or to reduce the computational load. Furthermore, it is suggested to avoid landmarks in

the route instructions which are not fixed in time or place. We realized we need to take into account the agent's point of view to successfully navigate him. Moreover, it is recommended to merge metric and topological graph into weighted graph.

However, some challenges still remain for future work like the implementation of the cutting function, panoramic view of the agent, usage of transparent objects in 3D space and simplifications of the instructions based on visibility.

ACKNOWLEDGEMENT

For professional guidance, direction and assistance in making the thesis, I sincerely thank Univ. Ass. Dipl.-Ing. Dr.-Ing Paolo Fogliaroni.

I also thank my family, Tomaž, co-workers and all those who supported, encouraged me in the time of the study and provided the unforgettable moments.

CONTENT

LIST OF TABLES

LIST OF EQUATIONS

LIST OF FIGURES

LIST OF EXAMPLES

ABBREVIATIONS

AL	Above Left
AR	Above Right
BL	Bottom Left
BR	Bottom Right
GNSS	Global Navigation Satellite System
IR	Infrared
J	Joint
L	Left
LOS	Line Of Sight
LZ	Light Zone
OC	Occlusion Calculus
R	Right
RCC	Region Connection Calculus
RFID	Radio Frequency Identifier Description
ROC	Region Occlusion Calculus
SZ	Shadow Zone
TZ	Twilight Zone
TZJ	Twilight Zone Joint
TZL	Twilight Zone Left
TZR	Twilight Zone Right
USID	Ultrasound identification
VCO	Visual Cyclic Ordering
VSO	Visual Sphere Ordering

1 INTRODUCTION

We use navigation systems on a daily basis, especially when exploring new environments to successfully arrive at our final destination. At the moment, most of navigation systems are mainly developed for the outdoor environment making the indoor navigation system somehow neglected yet the need for it is steeply rising. There already exist some techniques for navigating in indoor environment, however they have drawbacks. Therefore, we will analyse different method, which might be a better suited option to localize and navigate people.

1.1 Outdoor and indoor navigation

Processes and techniques for positioning and navigation have been quite standardized for the outdoor navigation with the usage of well-established system – Global Navigation Satellite System (GNSS). Because of well distributed satellites around the globe, this technology can be used almost everywhere except indoors, since the signal cannot penetrate through the walls of the buildings. Therefore, different techniques were developed for indoor environments. Some of these try to replicate the GNSS triangulation technique by replacing satellites with other signal distributors like Wireless Local Area Network (WLAN), Radio Frequency Identifier Description (RFID), Infrared (IR). Others try to avoid determining the unknown position by detecting three points with known positions and instead localize a user by detecting only one of opportunely distributed tags (RFID, Bluetooth, barcodes) in the environment. Another type of approach tries to build movement trajectories from sensors available in modern mobile devices such as compasses, magnetometers, and accelerometers. Other methods like fingerprinting and computer vision require a comparison between the samples of previously performed measurements, which are stored in a database, and current measurements carried out by the user. His position is extracted when a match is found (Fallah et al. 2013).

However, all of these techniques have drawbacks, RFID, IR, Bluetooth require installation and usage of extra devices in the environment and have, because of that, high installation and maintenance costs. Negative part of mobile sensors is low accuracy, whereas of WLAN is a multipath propagation. One of the problems of computer vision and fingerprinting is creation of a database.

Therefore, a different technique should be developed, which would avoid these disadvantages, a method which would effectively navigate the user.

1.2 Cognitively adequate methods

Previously mentioned techniques localize people based on metric coordinates and precisely track user's position while navigating. However, there exist other approaches that utilize a qualitative way of localization, such that divide space into regions. Created regions can be distinguished based on observation of distinctive visual events, i.e. landmarks (Levitt and Lawton 1990). "Landmarks form cognitive anchors, markers, or reference points for orientation, wayfinding and communication" (Richter and Winter 2014:1). They visually stand out from surroundings, or have a significant meaning, so that people easily notice them. In cognitively adequate methods qualitative coordinates are obtained, which can be described as what one can see at a certain position, i.e. which objects are visible from a certain position.

1.3 Motivation and goal of this work

A significant part of navigation is nowadays supported by electronic devices. They assist us, since the interpretation of the metric instructions (imagining where exactly to take a turn) is very difficult. The devices execute the wayfinding task for us usually the only thing we need to provide is a destination and the shortest route is calculated for us. Devices are used even during locomotion, providing constant updates of user's current position. This constant reliance on devices has a negative impact on spatial cognition abilities (Montello 2009; Richter, Dara-Abrams, and M. Raubal 2010). As a consequence of turn by turn instructions, people do not memorize the path of the travel. Instructions are cognitively inadequate, therefore people are just passively participating by listening to navigation instructions, instead of actively partaking in the navigation task (Burnett and Lee 2005).

In order to provide better support in orientation and facilitate user's comprehension, navigation based on landmarks is suggested. The routing instructions in this type of navigation are similar to a human description like "After crossing a bridge turn left" and not "After 100 m turn left". In this example, a bridge represents a landmark.

In this thesis, we explore and analyse a different solution to navigating people in indoor environments. One that avoids problems in previously mentioned techniques and produces landmark-based instructions. The idea is to navigate people based on qualitative coordinates based on a visibility. So, if one wishes to observe an entity, it is not only important that one is in its vicinity, but also that a third entity does not obstruct one's view. For this reason, visibility is a spatial property that can be considered as the relation between three objects: the observed object, the observer and the obstacle.

Based on these relationships, it is possible to calculate visibility between objects in the environment. However, the main requirement to exploit this technique is to have a detailed database of the environment. With that we divide space into regions, i.e. extract qualitative coordinates.

Example 1: Landmark-based route instructions

By using regions, we can create landmark-based instructions such as: **"walk around A keeping it on your right until you start perceiving B on A's left".** Landmarks in this case are objects A and B that could be replaced with the real objects from the environment like a desk information centre and a staircase.

There already exist qualitative localization and navigation system based on visibility information in 2D space proposed by Fogliaroni et al. (2009). However, the proposed approach does not provide enough information to navigate people, since we perceive the environment in three dimensions. Therefore, a visibility is part of a 3D space. For this reason, we analyse the extension of Fogliaroni et al. (2009) in 3D space, with its framework being developed in Fogliaroni and Clementini (2014). We present the main challenges that need to be tackled in order to get a functional implementation of such a landmark-based navigation system based on 3D visibility information, and address a subset of those. Some challenges concern the complexity of the system and are caused by the transition from 2D space to 3D space. Other challenges are more related to simplification of computation, information overload and navigation.

1.4 Outline of the work

The remainder of this work is structured as follows. We first discuss the background and related work in Chapter 2, which starts with cognitive aspects of navigation and continues with description of possible indoor navigation and different visibility models. Next, the visibility based navigation in 2D is presented, continuing with the description of the 3D visibility model in detail.

Based on those frameworks, visibility-based navigation is derived in 3D space and supported with some examples in Chapter 3.

Chapter 4 focuses on cognitive and complexity related issues that appear in 3D. There we define which objects from the environment are needed for the production of instructions presented in Example 1. We propose a special treatment of some indoor elements. The fact that some objects can be moved and are not constantly located in the same position was also discussed. Moreover, we are considering an agent's point of view and proposing a combination of metric and topological graph.

Chapter 5 concludes the thesis and describes other issues that are left for future work.

2 BACKGROUND AND RELATED WORK

This chapter will give an overview of related topics that are in conjunction with the thesis. In Section 2.1 we present a cognitive view of how people perceive space and if there is any difference between the orientation in the outdoor environment and the indoor environment. We will continue with examples of how to produce route instructions, which are based on landmarks and not on metric distance. End of Section 2.1 concludes with the criteria, that describes the characteristics of landmarks in more detail. Next Section 2.2 will focus on indoor navigation, where we will give first an overview of different techniques in indoor navigation that are currently available. Secondly, we further outline to what landmarks people in the indoor environment most commonly refer to. As has already been explained, an alternative to existing techniques on indoor navigation is visibility-based navigation. The basics of it are presented in Section 2.3, which represents different visibility models, classified into two groups. At the same time, we explain why a specific group is more appropriate for us. Section 2.4 reviews the usage of a visibility model for localization in 2D. Finally, Section 2.5 discusses the extension of the visibility model to 3D space, which will serve as a base to extend the localization and navigation approach to 3D.

2.1 Cognitive aspects of navigation

2.1.1 Spatial Cognition

The role of landmarks in human perception and memory is researched in the field of cognitive science. Aim of Spatial cognition science is to research human's as well as animal's mental representation, perception, memorization, interaction with the environment's spatial characteristics (Waller and Nadel 2013). Montello (2001) claims that spatial features and objects like landmarks influence on memory such that they configure mental spatial representations of them. It does not necessarily mean people need to be directly in the environment and move in it to gain spatial knowledge; it is also possible to acquire knowledge indirectly, by maps, images, and language.

Montello and Raubal (2013) marked wayfinding as a crucial function of spatial cognition and is described as one of the sub processes undergoing navigation, together with locomotion (Montello 2005) and spatial orientation (Wang and Spelke 2000):

- *Wayfinding* could be described as path planning; to reach the destination based on some tactic and strategy i.e. cyclist could plan a shortest path or a safest path.
- *Spatial orientation* is the ability to imagine the locations of a person relative to the environment i.e. to be aware where are we located with respect to the destination and other objects during the movement.
- *Locomotion* is an actual movement when executing the path (Richter and Winter 2014:48–50).

Siegel and White (1975) found that any prominent occurrence during locomotion provokes a memory located in space. These memories are called landmark knowledge, which could be also described as knowledge of distinctive objects or scenes stored in memory (Montello 1998).

Golledge (1999) analysed which path criteria are most frequently used in wayfinding process:

1. shortest route - shortest route based on the distance,
2. quickest route - shortest route based on time,
3. route of fewest turns - route which requires least number of decision points,
4. most aesthetic route - route which is aesthetically attractive to the user,
5. first noticed route - route based on the strategy which minimizes the time of identification,
6. longest-leg-first route – route with the longest first segment, before the decision point.

The findings of this study showed that all of them, even the first two, are connected to landmarks. In the past, it was already established that there exists a correlation between number of landmarks along the route and distance evaluations in space or time (Sadalla and Magel 1980; Sadalla and Staplin 1980; Thorndyke 1981). Route of fewest turns represents a path with least number of landmarks, whereas most aesthetic route could be a path passing a number of landmarks. First noticed route could represent a path with strongest salient landmarks and the longest-leg-first route could start out with a single landmark ahead and other information would be added later (Richter and Winter 2014:52).

Landmarks are defined as natural, built or culturally shaped features that stand out from their environment (Golledge 1999), that are unique or in contrast with their neighbourhood (Siegel and White 1975).

Several studies confirmed the importance of landmark-based navigation. Michon and Denis (2001), for example, carried out two studies. In the first one, the participants generated route instructions after they travelled it. They discovered that participants used landmarks to produce directions, especially at decision points. In a second study, people were guided by minimal navigational instructions and afterwards asked to improve route directions. Participants considered landmark as an important part of route instructions. Tom and Denis (2003) investigated on the difference between landmark and street names route directions. They concluded that landmark-based instructions are more efficient than street name-based directions and that people provide more landmarks than street names when giving instructions.

2.1.2 Spatial orientation and wayfinding

People use certain strategies and methods when they navigate themselves in the environment in order to successfully reach the destination. Allen (1999) suggested the strategies can be classified into four groups, in a case when user is located in a unknown environment:

- Oriented search: Initially users orient themselves in the right direction and systematically search the way until they reach the goal. The example of this method is arrows, which show a direction to the destination.
- Follow continuous marked trail: Users follow the route, marked by colour-coded trails or footprints.
- Plotting between landmarks: Users depend on instructions, which provide sequentially organized landmarks.
- Referring to a cognitive map: Users navigate themselves based on their mental map of the environment (Vilar, Rebelo, and Noriega 2014).

Lawton and Kallai (2002) proposed to divide wayfinding strategies only in two main groups – orientation strategies and route strategies. Orientation strategies use reference points which are globally available (i.e. seen from different positions in the environment, like sun, tall buildings). Based on them user maintains a sense of his position in relation to a reference points. Route strategies exploit a certain route to travel from one position to another (Hund and Minarik 2009; Hund and Padgitt 2010).

Previously mentioned proposed strategies were mostly investigated in outdoor environment, but wayfinding and spatial orientation in indoor environments cannot be equalized with the outdoor, since people are more likely to get lost inside the buildings than outdoors, because the spatial orientation is harder to maintain inside (Hohenschuh 2004; Radoczky 2003). For instance, orientation strategy indoors in impaired, since the walls and the lack of windows prevent us to have a broader view.

Weisman (1981) concluded that the main factors influencing on indoor wayfinding are:

- visual access of different locations from a certain position inside the buildings
- architectural differentiation: which enables people to distinct between different locations based on their unique architecture
- floor plan complexity (i.e. arrangement of rooms and hallways)

More about the indoor navigation will be discussed in Section 2.2.

2.1.3 Landmark route instructions

People are quite familiar with metric-based instructions, since we are engaging with them on our daily basis. An example of metric instruction is: “After 200 m turn right.” The information of the location of a decision point i.e. where to change the direction is given by the meters. But the problem is that people difficultly estimate the travelled distance. They can imagine better if the decision point is presented by the landmark. In contrast to the metric-based instructions, landmark-based instructions are not as well known.

Raubal and Winter (2002) have also proposed a wayfinding instruction model, which includes landmark based instructions. According to the following scheme:

[AT landmark$_i$] +

[TURN LEFT | RIGHT | MOVE STRAIGHT] +

{ONTO street name} +

{(PASSING | CROSSING) landmark$_j$}$_{0...n}$ +

[UNTIL landmark$_k$].

The ‘[]’ represents required elements, ‘{}’ optional elements, ‘UPPER CASE TEXT’ language elements, ‘lower case text’ variables and $i \neq j \neq k$.

Duckham et. al. (2010) expanded the algorithm by considering several cases:

- If a landmark is located at a decision point: “[Perform action] onto [Street Name] at [Selected landmark]”
- If a route section is longer than a temporal threshold, an in-leg landmark needs to be included in instructions:
    - If there is no landmark at the decision point: “[Perform action] onto [Street Name] after [Selected in-leg landmark]”
    - Otherwise: “Continue [Action] the [Selected landmark]”, where
        - [Action] is “past” if the selected landmark has a point-based geometry shape
        - Otherwise [Action] is “along” if landmark is by the route or “through” if landmark overlaps the route

If no landmark is located at the decision point and the instructions were not produced by an in-leg landmark, then the instructions will contain distance information instead: “[Perform action] onto [Street Name] after [Distance] “.

2.1.4 Landmark characterization and extraction

In order to recognize and extract landmarks from within spatial database, their attributes need to be defined that measure the attractiveness of landmarks. Raubal and Winter (2002) presented an approach based on the attractiveness of landmarks, which focus on the outdoor landmarks. They distinguish between:

- *Visual Attraction* – landmark is significantly different from the surrounding objects: façade area, shape (proportion of height and width), colour, visibility (calculated in 2D), other visual properties (texture, condition etc.);
- *Semantic Attraction* – landmark has a meaning feature: cultural and historical importance, explicit marks (signs on the building like coffee house), other semantic properties, which are easy to recognize;
- *Structural Attraction* – landmark plays a major role or has a distinguished location in the spatial environment structure: nodes (like number of streets that intersect at a certain point), boundaries (landmark is more prominent the higher the energy that has to be spent to cross them, like rivers channel), other structural properties (regions like quartiers, districts etc.).

Caduff and Timpf (2008) chose a different approach to assess the salience of landmarks, with:

- *Perceptual Salience*: is extracted from Location-based Attention – which defines salient regions in spatial scene (colour, intensity, texture orientation), Object-based Attention – which focuses on the prominence of single objects or group of objects (size, shape, object orientation) and Scene Context – which takes into account the global visual scene instead of a single object (topology, metric refinements);
- *Cognitive Salience:* is based on Degree of Recognition – which describes how well is an object recognized by a subject, Idiosyncratic Relevance – which identifies objects significance to the observer (personal, cultural or historical);
- *Contextual Salience*: distinguishes between Task-based Context – considers the type of task that the user will execute (i.e. cross the bridge, along the riverside), Modality-based Context – which considers the mode of transportation (walking, driving, riding) and its influence on field of view (pedestrians field of view is includes all objects, whereas drivers have limited view, because of the traveling speed).

A major difference between the first Raubal and Winter (2002) and the second Caduff and Timpf (2008) approach is that the latter does not only consider perceptual attraction but is, additionally, context-sensitive. Raubal and Winter (2002) also proposed Equation (1) for computing total salience (S) value of landmarks:

$$S = w_v s_v + w_s s_s + w_u s_u \quad (1)$$

$$1 = w_v + w_s + w_u$$

Where s_v, s_s and s_u represent the visual, semantic and structural salience and w_v, w_s ad w_u are weights associated to them.

Klippel and Winter (2005) investigated better the structural component. They applied different scenarios, if a landmark is located on a walking link or not and if a landmark is located on a decision point. Structural salience takes value from 0 to 0.2, if landmark is not located on a walking link and it depends on the distance between the landmark, the nearest walking link of the suggested route and maximum visible distance of the landmark. If a landmark is located on a walking link, the structural salience is in range [0.2, 0.5] and it depends on the length of the walking link and on the distance of the landmark to the start and end nodes of the link. Values in the range [0.5, 1] are reached only if the landmark is located on a decision point. It depends on the distance to the decision point, and its visibility.

Duckham et al. (2010) focused on the calculation of the salience value, based on landmarks suitability and frequency of typical landmarks in a certain category. As suitability factors Duckham et al. extended Raubal and Winter (2002) work to a finer list of sub-characteristics:

- *Visual Character*: physical size, prominence, difference from surroundings, night-time and daytime salience, proximity to road;
- *Semantic Character*: ubiquity and familiarity, length of description;
- *Structural Character*: spatial extents, permanence.

Each landmark is ranked as: 'ideal,' 'highly suitable,' 'suitable,' 'somewhat suitable' or 'never suitable' for suitability and 'all typical,' 'most,' 'many,' 'some' or 'few' for frequency.

Fang et al. (2012) built an implementation relying on Duckham et al. (2010) and Klippel and Winter (2005). The visual salience value in their research was calculated with iLab Neuromorphic Vision[1], which basically computes salience map from an image as shown in Figure 1. The brightest pixels on the map represent the most salient areas, which stand out from their surroundings. A high value for a landmark describes a visual salience value of this landmark.

[1] http://ilab.usc.edu/toolkit/

Figure 1: Saliency map (right) computed from an image (left) (Fang et al. 2012).

Kennedy and Naaman (2008) have combined images analysis, tag data and image metadata to extract landmarks from datasets. Initially, the tags and location metadata are used to detect tags and locations that represent landmarks or geographic features. This step enables to speed up the process, since it reduces the number of images which are further analysed to extract representative sets of images for each landmark. In the last step, images are clustered in similar groups and linked, if they contain the same objects.

A similar approach was performed by Wither et al. (2013). They have first created data with Light Detection And Ranging (LIDAR), GPS receivers and cameras adequate to capture panoramas. Two pipelines were used for data processing: (1) building canonical image pipeline (BCIP) for providing a 3D view of landmarks and (2) text recognition pipeline (TRP) for sign and business detection. TRP was successfully applied for approximately 60 % of visible signs in panoramas others have unrecognizable font, inconvenient sign orientation or obstacles in front of them.

Tezuka and Tanaka (2005) proposed an entirely different method to extract landmarks: web mining approach. They did not focus on visual saliency instead their motivation relies on, how geographic objects are referred to by humans. Existing methods for text mining were adapted in a way that spatial context was considered, which improved the precision of extracting landmarks from web documents.

One of the latest proposals is from Wolfensberger and Richter (2015), where they implemented a mobile application for a user-generated collections of landmarks. The approach to use Volunteered Geographic Information (VGI) for landmark collection production was first suggested by Richter and Winter (2011) and Richter (2013). Their user-friendly application uses photo-based interface. To extract a landmark, one needs to take a photo of it to collect the user's position (via GPS) and heading (via the inbuilt compass). With this information, the area visible to the user is computed. The content of picture is not used in any way. Only sensor data are used to compute and rank landmarks and to presents them to the user. Afterwards any of the suggested landmark candidates needs to be approved by the user. Their intent is to provide a tool for the in-situ manual selection of landmarks.

2.2 Indoor Navigation

2.2.1 Indoor Navigation and Positioning Systems

In outdoor navigation one of the most important elements is Global Navigation Satellite Systems (GNSSs). It obtains device position with an accuracy that enables to navigate users through the environment. Unfortunately, this system cannot be used in an indoor environment because the accuracy decreases steeply, since the signals of GNSS are blocked by the buildings. Therefore, localization and navigation systems based on other methods and technologies need to be implemented. Currently, there are several possibilities for indoor positioning, which can be divided in four groups (Fallah et al. 2013):

- **Dead reckoning**
  Dead reckoning method uses data from several sensors (compasses, accelerometers, magnetometers and gyroscopes) located on a user to estimate his relative position with respect to a given origin. The absolute location of the origin needs to be obtained by a different technique. The disadvantages of dead reckoning are inaccuracy and the fact that a complementary method needs to be used (Fallah et al. 2013; Harle 2013).
- **Direct sensing**
  In direct sensing the position of a user is determined such that device on the user detects a single identifier or a tag, which has a known position.
  Widely used technology is *Radio Frequency Identifier Description* (RFID), where tags could be passive (they do not transmit signals) or active (Fallah et al. 2013). The drawback is that the signal cannot penetrate through the human body and if the environment is overcrowded, the method is not performing as well (Amemiya et al. 2004). Maintenance and installation of the tags are also costly.
  Infrared (IR) transmitters are also capable of locating a user. The problematic part of it is that the light interferes with IR and that the signal can only be detected within a certain range and line of sight, therefore a large amount of transmitters is required.
  The latter problem is also a drawback in *Ultrasound identification* (USID) and in addition to that, it is not as accurate because of the multipath propagation[2].

[2] Multipath is the propagation phenomenon that results in signals reaching the receiving antenna by two or more paths (Multpath Propagation Anon 2016).

Bluetooth beacons could impact negatively on a user's movement, because the device detection has a time delay. The disadvantage of Bluetooth beacons is also maintenance (Fallah et al. 2013).

To use *barcode* as a positioning method, one needs to scan them along the walking path to obtain the next navigation instructions. Each barcode encodes a unique ID which, in turn, represents a certain position in an environment (Mulloni et al. 2009). Even though the barcode installation does not require high environmental intervention and maintenance, it has a negative impact on navigation process since the user needs to find and scan the barcode (Fallah et al. 2013).

- **Triangulation**

  Triangulation is a method, which determines unknown position of a device based on three or more points with a known position. The position could be defined with (i) lateration, where distances are measured between the unknown point and known locations, or with (ii) angulation, where angles are measured instead of distances. Triangulation is the basic method used in GNSS, where points with known position are satellites (Hightower and Borriello 2001). In indoor the role of satellites is overtaken by other elements such as tags of ultrasound, Radio Frequency Identifier Description (RFID) or Infrared (IR) (Fallah et al. 2013). When the signals cannot penetrate through walls, Cell-Towers or WLAN can be used instead of satellites (Arikawa, Konomi, and Ohnishi 2007). The negative part of Cell-Towers and WLAN however is multipath propagation, which causes low accuracy (Arikawa et al. 2007; Retscher and Thienelt 2004).

- **Pattern recognition**

  In the order in which this method is utilized, the database of the environment needs to be established beforehand, so when the user needs to be located in the environment, the measurements are compared to the ones in the database. The position of the user is found, when the match is confirmed (Fallah et al. 2013).

  One of the pattern recognition techniques is the computer vision, where the user is navigated through the environment with a camera, where the current view is compared with the ones in the database. Because the images in the database have known position and orientation, these can be extracted when the match is found. Disadvantages are the high computational load during a matching process, different circumstances in reality compared to the ones in the dataset (e.g. light) and creation of the database. The latter is considered as a negative, because it requires space evacuation in order to record only static elements in the environment, so that people and other unwanted objects are not scanned and stored in the database. (Gerstweiler, Vonach, and Kaufmann 2015; Mautz 2012:4.2).

Signal distribution or fingerprinting technique compares the sets of signal measurements from user's position with a set of reference signals in the database. The accuracy depends on the coverage of samples in the database (Beal 2003).

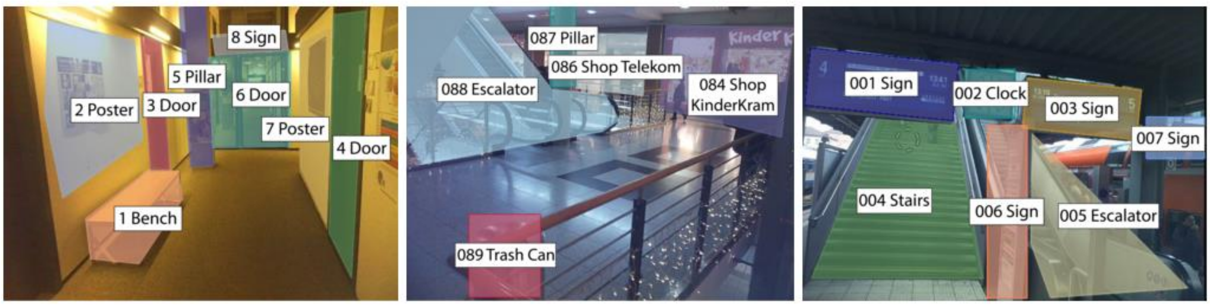
2.2.2 Indoor landmarks

In Section 2.1.4 characteristics of landmarks and the ways of extracting them from the surroundings were presented, but they mostly focused on the landmarks in outdoor environment. The question of which objects can be considered as an indoor landmark has been investigated in Ohm et al. (2014), where they have conducted a real world experiment with an eye tracking device in a complex university indoor environment. The authors were not only interested in where do people focus on, but also which objects are perceived as indoor landmarks. The conclusion was that one of the potential landmark candidates are elevators, escalators, stairs, doors, plants, information boards and signs, which can be divided into four main groups:

- architecture: pillars and fronts
- function: elements like doors, stairs and elevators
- information: objects like signs and posters
- furniture: objects like tables, chairs, benches and vending machines

Based on their experiment, the most referred landmarks are functional objects and the least used are furniture landmarks, whereas architecture and information objects are somewhere in between. The results (that the functional objects are preferred as indoor landmarks) have also been confirmed by Viaene et al. (2016).

Ohm et. al. (2015) expanded the investigation of what indoor landmarks are to other environments like shopping malls and train stations. The authors extended the landmark categories presented in Ohm et. al. (2014) to five groups, with the fifth being shops and restaurants. Their conclusion was that the best category group is changing depending on the environment. So in shopping malls the best category are shops, whereas in office buildings and train stations, the most referred landmarks are functional objects.



a) University. b) Shopping mall. c) Train station.

Figure 2: Examples of landmarks in different indoor environments (Ohm et al. 2015).

Sefelin et al. (2005) have focused only on train station environments and studied several methods (picture-based object recognition, think aloud protocols – path description, eye-catcher detection, picture based object description and naming, wizard of Oz prototyping) of finding suited landmarks for user navigation. One main finding was that, even if the most salient object is extracted, it is crucial how the landmark is described to the user in the instructions. For instance, instructions referring to the "Snack bar with the illuminated green sign" were more understandable than instructions referring the text written on the sign "Wettpunkt". On the other hand, the authors suggested that famous brands should always be described by their names.

In summary it can be concluded, that indoor landmarks are not always the same objects, but they change from an environment to another, because each building has its unique constitution and needs to be addressed differently.

2.3 Visibility Models

There exist several visibility models that could describe relations of objects in the environment. According to Fogliaroni et al. (2016) they could be combined in two main groups based on:

- primitive relations from an object-centred perspective (Galton 1994; Köhler 2002; Randell and Witkowski 2002; Randell, Witkowski, and Shanahan 2001)
- projective geometry properties (Fogliaroni et al. 2009; Fogliaroni and Clementini 2014; Tarquini et al. 2007).

Galton (1994) proposed **line of sight** -14 (LOS) calculus by combining region connection calculus - 8 (RCC) relations (Randell, Cui, and Cohn 1992) with two relative distance relations (*nearer* and *further*).

The calculus represents qualitative relations between the projections of two convex objects (A and B) as perceived from an observer's point of view. In case when A and B are freely moving, the observer could distinguish between the following 14 different relations (see also Figure 3):

C A is clear of B	JH A just hides B	EHI A is exactly hidden by B
JC A is just clear of B	JHI A is just hidden by B	F A is in front of B
PH A partially hides B	H A hides B	FI A has B in front of it
PHI A is partially hidden by B	HI A is hidden by B	JF A is just in front of B
	EH A exactly hides B	JFI A has B just in front of it

For example, if the projections of A and B as perceived from observer's viewpoint are *equal*, the two possible relations may hold:

1. if A is *nearer* to the observer than B: A *exactly hides* B (relationship EH in Figure 3),
2. if A is *further* to the observer than B: A is *exactly hidden* by B (relationship EHI in Figure 3).

In case when projections of A and B are *disconnected* or *externally connected*, then the LOS calculus does not differentiate the relationship between the nearer and further relative distance. So when the objects are disconnected, the relation *is clear of* holds and when they are externally connected, the relation is described as *is just clear of*.

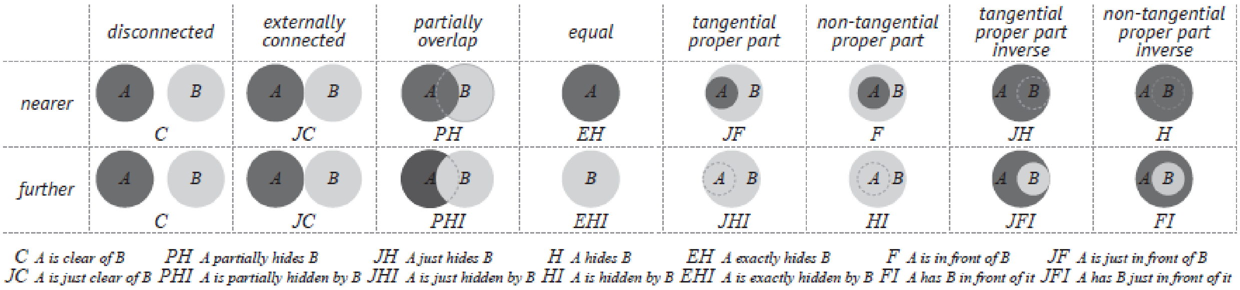


Figure 3: The 14 line of sight relations from Galton (1994), which were extended from region connection calculus-8 (RCC). Image from Fogliaroni et al. (2016).

Randell et al. (2001) and Randell and Witkowski (2002) extended Galton's calculus to **region occlusion calculus** - 20 (ROC) with 20 relations between objects, by considering convex and concave objects (cmp. Figure 4). ROC preserves all 14 relations from LOS and adds 6 additional that describe mutually occluding relations that can only hold between non-convex objects. For example, in the relation *MutuallyOccludesNTPP* (cmp. Figure 4) the lighter object simultaneously occludes and is occluded by the darker object.

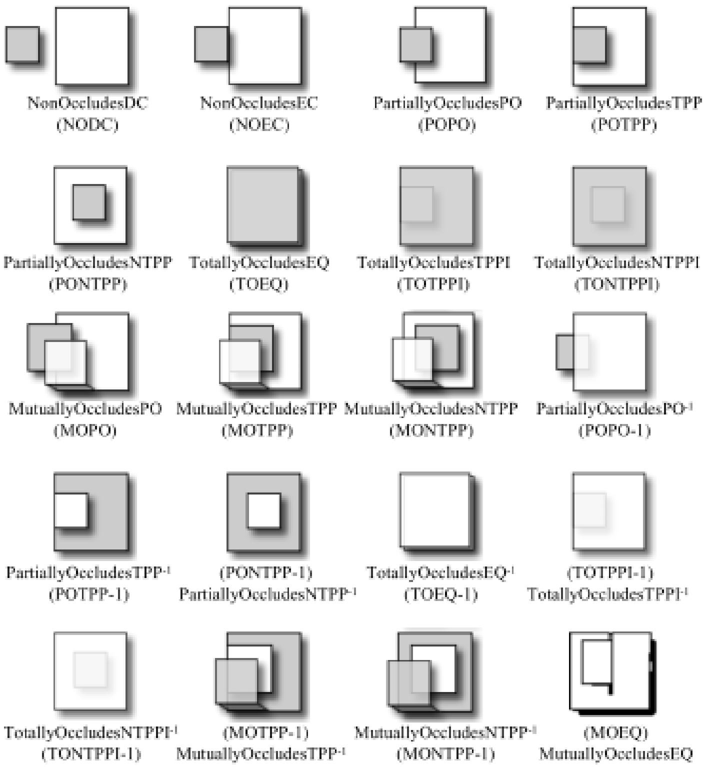


Figure 4: The 20 relations of region occlusion calculus from Randell et al. (2001). Image from Randell and Witkowski (2002).

Köhler (2002) followed Galton’s LOS idea and combined it with point algebra basic relations and RCC-5 (derived from RCC-8, such that boundaries of objects are ignored). Köhler (2002) proposed **occlusion calculus-8** (OC) consisting of the 8 relations presented in Figure 5. In comparison with LOS, relations, where objects are touching on borders, are being ignored in OC.

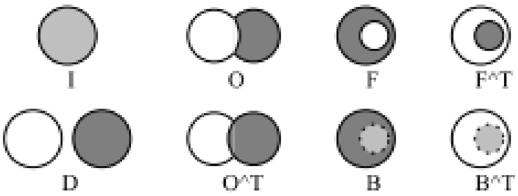


Figure 5: The 8 relations of occlusion calculus from Köhler (2002).

Tarquini et al. (2007) choose to select a different approach with the properties of projective geometry, where the outcome of the visibility model is a set of 7 ternary relations between regions, where A is the observed object, B is an obstacle and C the observer. In order to extract the relations, the internal and the external tangents between objects B and C, which describe acceptance areas of visibility – Shadow Zone (SZ), Twilight Zone (TZ), Light Zone (LZ) visualized in Figure 6 – need to be outlined first.

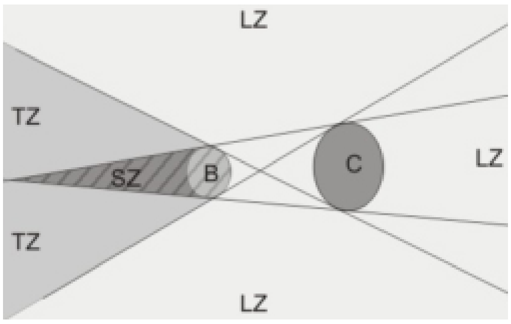
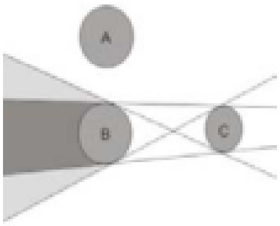
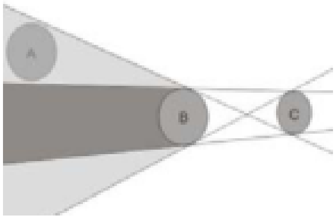


Figure 6: Acceptance area of visibility outlined from external and internal tangents (Tarquini et al. 2007).

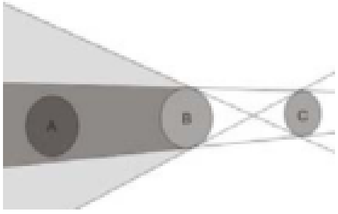
Based on those areas, 7 different relations described in Figure 7 are defined. If an object A is located only in LZ, then the relation is called *Visible* (cmp. Figure 7(a)). In case it is positioned only in TZ like in Figure 7(b), it is *PartiallyVisible* to C. *Occluded* relation corresponds to the SZ. All these three relations examples are single-tile relations, because object A is located in only one zone. If an object A intersects more zones simultaneously then we are dealing with multi-tile relations. Multi-tile relations are named after merging the single-tile relations corresponding to the zones. For example, object A intersects LZ and TZ in Figure 7(d), therefore the corresponding relation is *Visible&PartiallyVisible*.



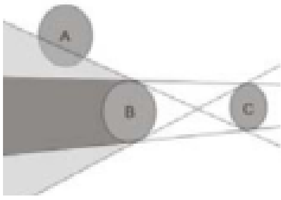
(a) Visible.



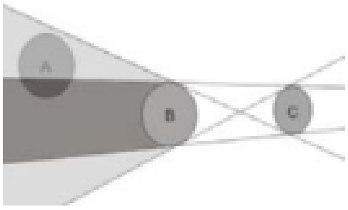
(b) PartiallyVisible.



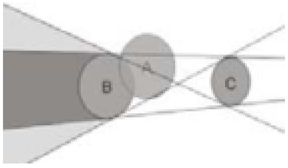
(c) Occluded.



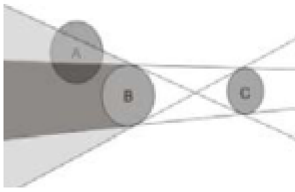
(d) Visible&PartiallyVisible.



(e) PartiallyVisible&Occluded.



(f) Visible&PartiallyVisible&Occluded.



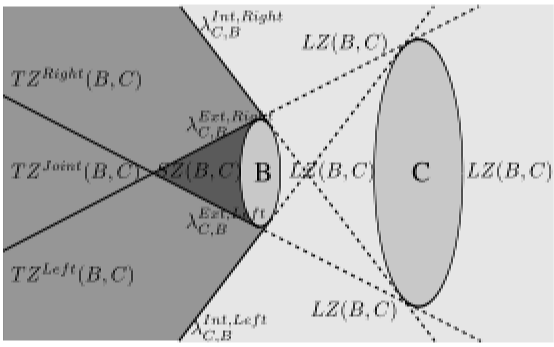
(g) Visible&PartiallyVisible&Occluded.

Figure 7: Examples of 7 visibility relations, which are based on the acceptance area of visibility (Tarquini et al. 2007).

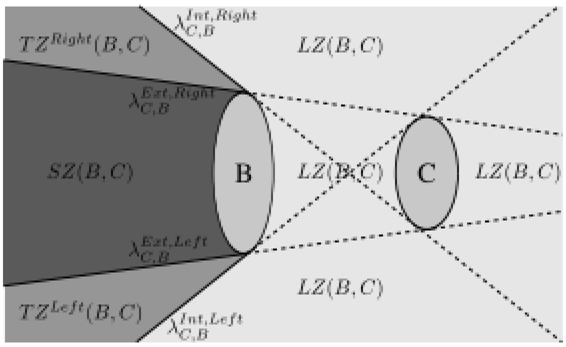
A more extensive approach is covered in Fogliaroni et al. (2009), where a finer space partition is implemented. First, they separate TZ into left (TZL) and right (TZR) (Figure 8). Secondly, they consider the case when the external tangents are intersecting each other behind the obstacle (Figure 8(a)) and define it as Twilight Zone Joint (TZJ). Due to those improvements, new relations are acquired: *PartiallyVisibleLeft*, *PartiallyVisibleRight* and *PartiallyVisibleJoint*. The resulting visibility relations are:

$$\begin{gathered}
\text{Visible } (A, B, C) \Longleftrightarrow A \in \text{LZ } (B, C) \\
\text{PartiallyVisibleLeft } (A, B, C) \Longleftrightarrow A \in \text{TZLeft } (B, C) \\
\text{PartiallyVisibleRight } (A, B, C) \Longleftrightarrow A \in \text{TZRight } (B, C) \\
\text{PartiallyVisibleJoint } (A, B, C) \Longleftrightarrow A \in \text{TZJoint } (B, C) \\
\text{Occluded } (A, B, C) \Longleftrightarrow A \in \text{SZ } (B, C)
\end{gathered} \tag{2}$$

As an addition they have utilized this qualitative way of reasoning and presented its usage in localization and navigation, which is explained in detail in Section 2.4.



(a) Limited shadow zone.



(b) Unlimited shadow zone.

Figure 8: A detailed example of acceptance areas defined in Fogliaroni et al. (2009).

Fogliaroni and Clementini (2014) proposed the usage of the visibility model in 3D as an extension of 2D model. Not only that they have also included non-convex objects in their theory and argued for the usage of the model in human wayfinding. The model is explained in more details in Section 2.5.

When comparing the first group of visibility models (Galton 1994; Köhler 2002; Randell and Witkowski 2002; Randell et al. 2001) and the second group (Fogliaroni et al. 2009; Fogliaroni and Clementini 2014; Tarquini et al. 2007) it becomes clear that the latter group is founded on more extensive grounds. Not just because the point of view is not seen as a point but as a convex region, or the usage of visibility relation modelled as ternary relation contributing to a more general model (Tarquini et al. 2007), they also allow for both egocentric and allocentric representation (Fogliaroni et al. 2009), whereas the models from the first group do not. Moreover, the second group also handles different types of observed objects, no matter the size (it could be a point or a more spacious object) or geometry (it could be convex or non-convex). Most importantly, it supports visibility model in 3D space, which is one of the most important things if we want to use it in navigation.

2.4 Visibility-based navigation in 2D

Fogliaroni et al. (2009) introduced the usage of a their visibility model (presented in Section 2.3) in the navigation. This approach utilizes a qualitative way of localization, such that divides space into regions. Created regions can be distinguished based on observation of distinctive visual events, i.e. landmarks (Levitt and Lawton 1990). By applying this method qualitative coordinates are obtained, which can be described as what one can see at a certain position, i.e. which objects are visible from a certain position.

First Fogliaroni et al. (2009) introduced some simplifications of visibility model. When navigating a person, we are dealing with a point-region case, where the observer object (c) can be seen as a point and an obstacle (B) as a region. Due to this simplification the internal and external tangents will coincide. As a result, partly visible relations and Twilight Zones will not appear anymore (cmp. Figure 9(a)). From the Equation (2) the resulting influence rules are obtained:

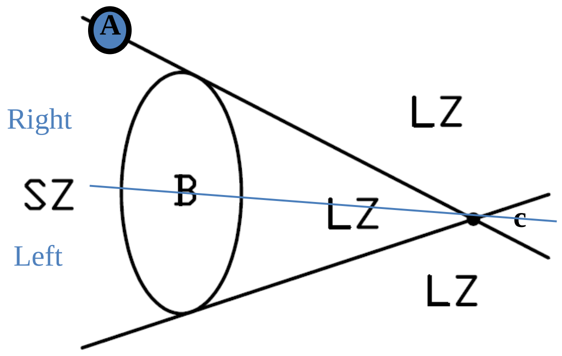
$$\begin{gathered}\text{Visible}(A, B, c) \Longleftrightarrow \text{Visible}(c, B, A)\\ \text{Occluded}(A, B, c) \Longleftrightarrow \text{Occluded}(c, B, A)\\ \text{Visible} \wedge \text{Occluded}(A, B, c) \Longleftrightarrow \text{PartiallyVisible}^{*}(c, B, A)\end{gathered} \tag{3}$$

* either Left, Right or Joint.

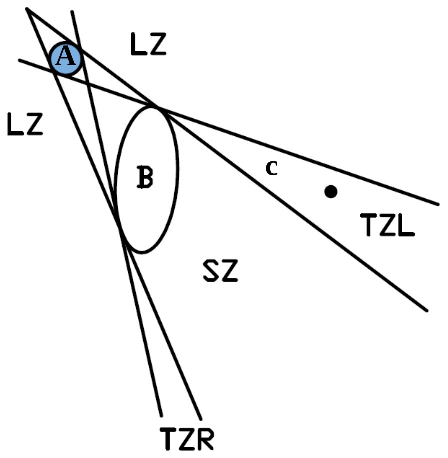
In Equation (3) is presented a transition between perspective i.e. transition from (A, B, c) to (c, B, A). Relations *Visible* and *Occluded* are not changing with regard to the permutation of first and third objects. However the attention must be paid to the relation *Visible ∧ Occluded*(A, B, c), since the orientation also changes. Imagine a line between the observer c and a centre of object B (blue line in Figure 9(a)) which divides a space on Left and Right side (from observer's perspective). Say only a part of A is visible from c then the following properties apply:

$$\begin{gathered}A \in \text{Right} \wedge \text{Visible} \wedge \text{Occluded}(A, B, c) \Longleftrightarrow \text{PartiallyVisibleLeft}(c, B, A)\\ A \in \text{Left} \wedge \text{Visible} \wedge \text{Occluded}(A, B, c) \Longleftrightarrow \text{PartiallyVisibleRight}(c, B, A)\\ A \in \text{Right} \wedge A \in \text{Left} \wedge \text{Visible} \wedge \text{Occluded}(A, B, c) \Longleftrightarrow \text{PartiallyVisibleJoint}(c, B, A)\end{gathered} \tag{4}$$

An example of relation *PartiallyVisibleLeft*(c, B, A) from Equation (4) is depicted in Figure 9.



(a) Relation *Visible ∧ Occluded*(A, B, c), where c perceives A on the right side of B.



(b) Relation *PartiallyVisibleLeft*(c, B, A), where A perceives c on the left side of B.

Figure 9: An example of possible visibility relation in point-region case.

Based on the visibility model, Fogliaroni et al. (2009) proposed the usage of a qualitative coordinate system for navigation, where qualitative coordinates are represented by visibility regions (i.e. what can one see from a certain position). The idea is explained in 2D space (Figure 11), where the plane is divided into regions in such a way that between every pair of objects, internal and external tangents are calculated. These are the defining visibility relations. It needs to be emphasized that the entire line is not included in subdivision, instead only a half-tangent of it is used. Figure 10 is an example of half-tangents between two objects highlighted in black colour. They are extracted so that the segments between the objects are deleted (highlighted in red).

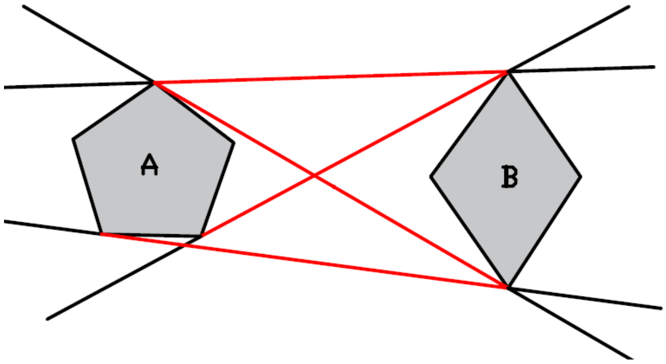


Figure 10: Half-tangents (black colour) between the two objects, which are extracted from internal and external tangents (black and red colour).

During this process it is necessary to check if:

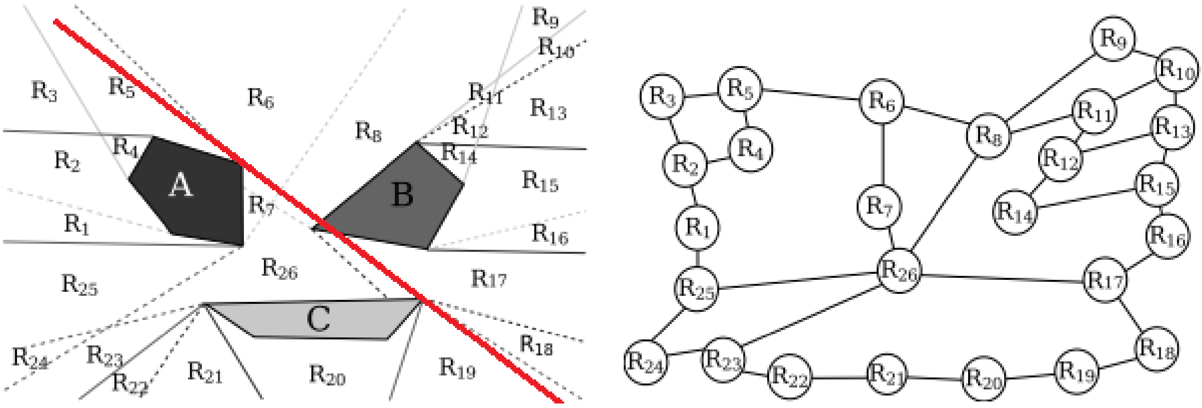
1. the tangent is intersecting any other object between the two points that define the tangent,
2. tangent cut through the objects.

If the first case is true, then this tangent needs to be ignored and cannot represent region boundary. This event is presented in Figure 11(a), where the red line illustrates external tangent between object C and A on the right side (as going from C to A). It is visible that object B is blocking the tangent therefore the tangent is not outlining a boundary between the regions.

All constituted tangents need to undergo the second inspection, where it is important that the constituted tangents do not cut through the objects, but instead they are cut by the object's boundary. This was performed on an internal tangent between object A and B, dividing regions R_{26} and R_{17} in Figure 11(a). These two verifications are performed by the so called *cutting function*.

From the plane regions, a topological map (Figure 11(b)) can be established as an undirected graph, where:

- each node represents a region from the planar subdivision
- an edge exists between nodes if there is the possibility of moving between the corresponding regions by crossing only one boundary line.

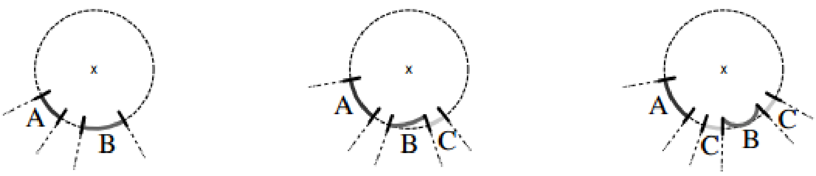


(a) Plane subdivision (b) Topological map

Figure 11: Plane division in regions and corresponding topological map depended on visibility model (Fogliaroni et al. 2009).

The obtained regions in Figure 11 represent qualitative coordinates that describe which objects are visible from a certain region. The regions are obtained by merging visibility reference frames from every pair of objects. Consequently, certain visibility relations have to hold between a point in a region and every object. By using the properties of Equations (3) and (4) it is possible to deduce which relations hold between the agent and all objects.

In order to navigate through the plane, one needs to be able to distinguish between the regions. It could happen that the identical visibility relation occurs in different regions, therefore visibility relations itself are not enough. For these reason the authors proposed to use Visual Cyclic Orderings (VCOs), which describe the cyclic order in which obstacles are perceived from a certain point (Figure 12).



(a) $\langle A\,B \rangle$ (b) $\langle A\,C^{C}B \rangle$ (c) $\langle A\,C^{C}B^{C}\,C \rangle$

Figure 12: Cyclic order of objects perceived by an agent (Fogliaroni et al. 2009).

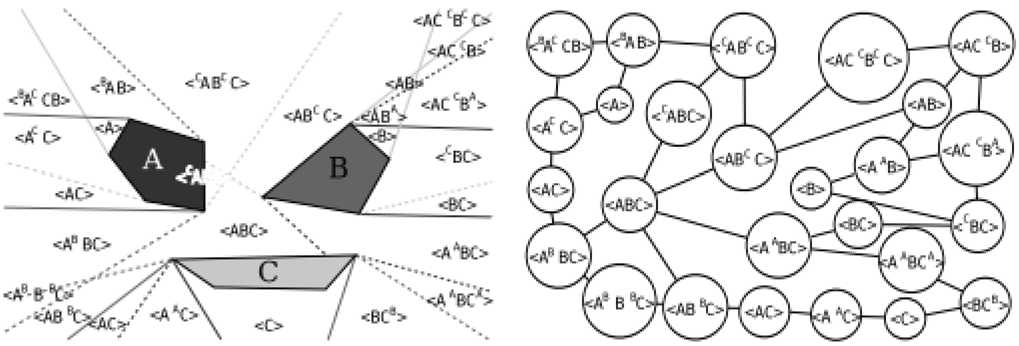
An example of a VCO label in Figure 12(a) corresponds to a visibility relation if an agent is standing in region R_{11} (Figure 11(a)). When exploiting five different possible relations (Visible, Occluded, Partly Visible Right, Partly Visible Left, Partly Visible Joint), which are discussed above in this chapter, it becomes evident that the objects A and B are visible to the agent, whereas C is occluded. VCO in Figure 12(a) perceives space between A and B, where it can be seen that the objects are completely visible, hence the label <AB>.

If an agent is standing in a region R_{10}, A and B are visible, whereas C is only partly visible (Figure 12(b)) because it is occluded by B, which means B is closer (this is shown in the image as an indent bordering towards the centre). It is seen that C is on the left side of B. A ^{C}B in VCO label is denoting an occlusion, where to the occluding object (in this case B), the occluded object (in this case C) is added as a superscript on its respective side.

It is also possible that the occluded object appears on both sides of the occluding object, which means the agent is located in JoinTwilightZone. This situation occurs in R_9, where the agent can see the C on the left and the right side of B, so the superscript is on both sides of the letter ($^{C}B^{C}$) and the borders of B are both indent towards the centre. The corresponding VCO is described in Figure 12(c).

Figure 13(a) shows the plane subdivision, with the corresponding VCO labels. Through the VCO labels, which are determined for each region separately, we can distinguish between regions and identify the position of the agent. Nevertheless, the regions are still not completely different from each other and it may still appear that the regions have the same VCO labels. This special occurrence may be seen in Figure 13(a), where <AC> label appears twice. The authors proposed a solution considering the VCOs two neighbouring regions to resolve the ambiguity.

A topological map can be derived from a geometric representation of the environment, where nodes are labelled with the corresponding VCO as seen in Figure 13(b).



(a) Plane subdivision. (b) Topological map.

Figure 13: Computed VCO for every region (Fogliaroni et al. 2009).

- **Navigation and path planning**

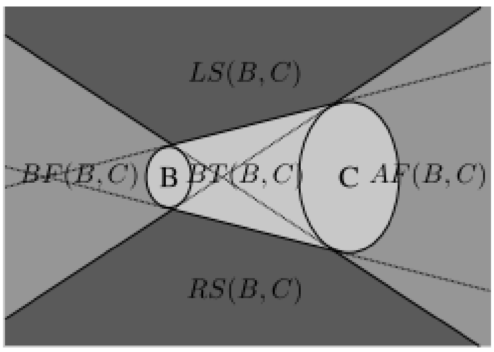
Fogliaroni et al. (2009) discussed the execution of path planning and navigation based on topological map of the environment and suggested to use a Dijkstra's algorithm (Dijkstra 1959) or A* (Hart, Nilsson, and Raphael 1968) to compute the shortest path between nodes in a graph.

An agent can only travel from the start to the end, so that it travels through the nodes, which are interconnected by the edges. As already mentioned above, each edge represents the boundary between the two regions, which means that the difference between these two regions depends on what kind of a half-tangent the boundary belongs to. So if a half-tangent is separating Shadow Zone and Right Twilight Zone (regions <BC> and <A^{A}BC>), that means it belongs to the right external tangent. When one travels from Shadow Zone to Right Twilight Zone, an object becomes partly visible on the left of B and if the direction is reversed (from Right Twilight Zone to Shadow Zone), the object disappears. Due to a finite number of regions and with it also boundaries, consequently there is a finite set of crossing cases, since one can define two navigation behaviours for each boundary, like discussed above.

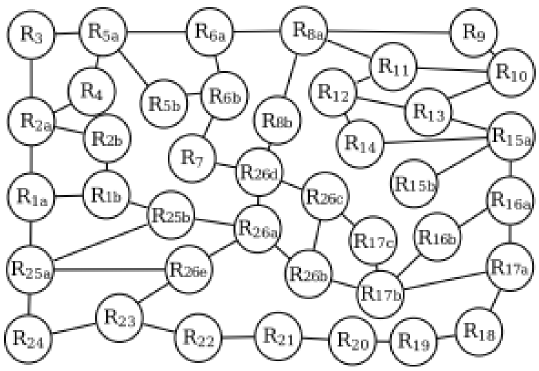
Therefore, it is possible to produce natural language instructions like “walk around B keeping it on your right until you start perceiving A on B’s left”, or “walk around B keeping it on your left, until A disappears from your sight” (Fogliaroni and Clementini 2014).

- **Finer plane subdivision**

Fogliaroni et al. (2009) realized that the Light Zone is not partitioned enough and so there may be situations where a specific region is too large. For this reason, it was proposed to upgrade to a more detailed partition of the regions (Figure 14(b)), where they take into account the work of Billen and Clementini (2004), Clementini and Billen (2006) which created the 5 intersection model presented in Figure 14(a). So Fogliaroni et al. (2009) combined 5 intersection model with the exeptance areas depicted in Figure 8 and as a result gain finer topological map, which obtains 14 more regions than in the previous subdivision presented in Figure 11.



(a) The 5 intersection model.



(b) Finer topological map.

Figure 14: Finer topological map created based on 5 intersection model and visibility model (Fogliaroni et al. 2009).

2.5 3D visibility model in detail

The scope of this thesis is to navigate people, therefore a previously discussed navigation in 2D is insufficient and can be used only as a theoretical background for the further navigation extension into 3D environment.

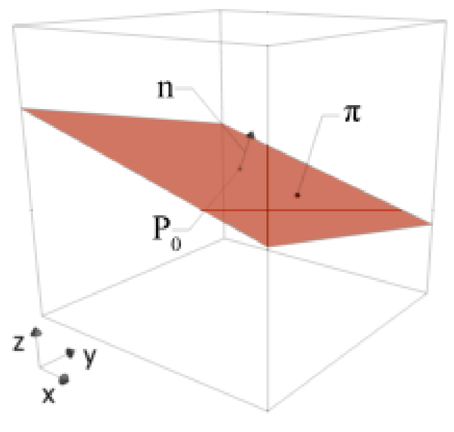
2.5.1 3D relative direction and visibility model

The principle of 3D visual model, proposed by Fogliaroni and Clementini (2014), is the same as in 2D. There is an observed object A and the two reference objects (B and C), the difference is that a direction (vector d) needs to be defined in order to obtain ternary relations. The latter could be defined as the gravity direction or, in alternative, as a vector pointing from feet to head.

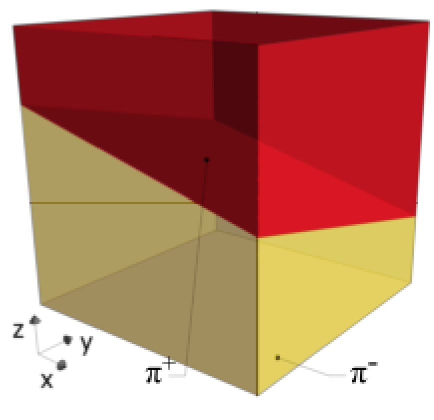
In 3D external and internal tangent planes could be defined between B and C, similar as in 2D, where the external and internal tangent lines were defining the acceptance areas (cmp. Figure 8)

In order to continue, some elements of 3D Euclidean space $\mathbb{E}^3$ need to be explained:

- **Plane $\boldsymbol{\pi}$:** Is a flat surface defined by a point P and normal vector n (cmp. Figure 15(a)).
- **Half-space:** Is obtained, when a plane divides $\mathbb{E}^3$ into a positive (π^+) and negative (π^-) Half-spaces. Direction of a normal vector is defining the positive half-space (cmp. Figure 15(b)).



(a) Point-vector definition of a plane $\boldsymbol{\pi}$ by point $\boldsymbol{P}$ and normal vector $\boldsymbol{n}$.



(b) Half-spaces obtained from a plane $\boldsymbol{\pi}$.

Figure 15: Definition of a plane and half-spaces (Fogliaroni and Clementini 2014).

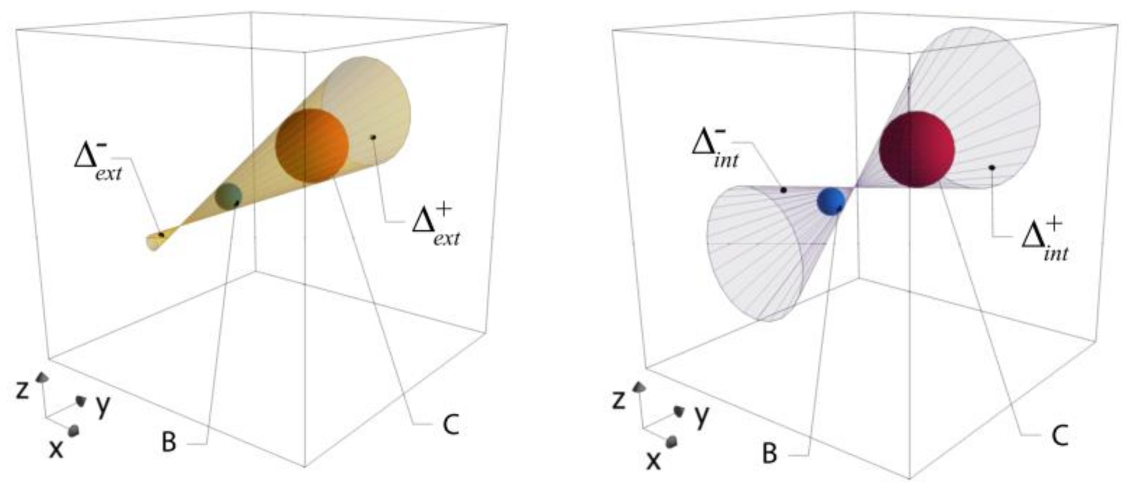
Assuming that both reference objects B and C are spheres, therefore there exist an infinite set of planes Π which are tangential to the spheres. Meaning that each plane has one common point with C and one common point with B. Imagine that each plane is defined by the common point with C and a normal vector, directed from this point to the centre of C. Based on that, we obtain half-spaces, such that C always falls in π^+. The tangential planes Π can be split in two sets:

- **external tangent planes Π_{ext}**: is the set of all planes π such that other reference object B is also contained in π^+.
- **internal tangent planes Π_{int}**: is the set of all planes π such that other reference object B is contained in π^-.

From the **Π_{ext}** it is possible to derive the pair of **external cones** ($\Delta^+_{ext}, \Delta^-_{ext}$) presented in Figure 16(a). The intersection of the positive half-spaces π^+ form the positive cone Δ^+_{ext}, whereas the intersection of negative half-spaces π^- denote the negative cone Δ^-_{ext}.

Similarly from the **Π_{int}** it is possible to derive the pair of **internal cones** ($\Delta^+_{int}, \Delta^-_{int}$) (cmp. in Figure 16(b)). The intersection of the positive half-spaces π^+ form the positive cone Δ^+_{int}, whereas the intersection of negative half-spaces π^- denote the negative cone Δ^-_{int}.

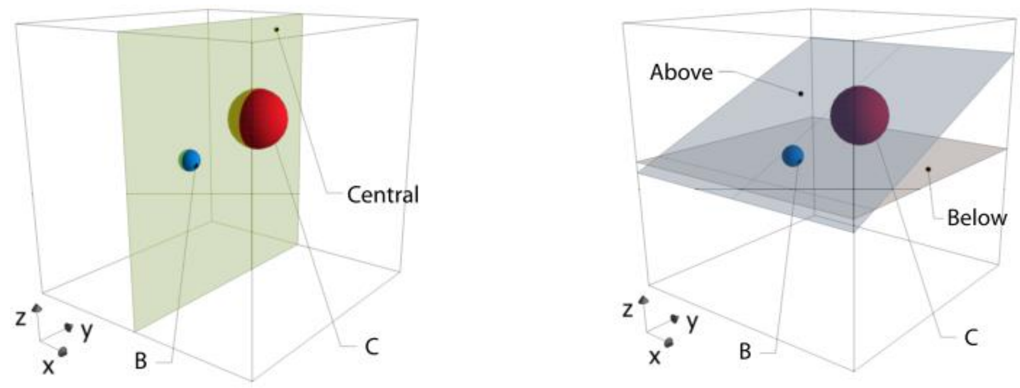
When combining external and internal cones, one obtains Figure 18(a).



(a) External cones. (b) Internal cones.

Figure 16: Computed cones from internal and external tangents (Fogliaroni and Clementini 2014).

Fogliaroni and Clementini (2014) also defined three planes depicted in Figure 17, which help to outline the accaptance volumes. **Plane *Central***, which goes through the centroids of B and C and is parallel to the vector d, as depicted in Figure 17(a). With this plane, the left and the right sides are obtained. **Planes *Above* and *Below*** are defined as the external tangent planes, where plane Above passes through the max value and plane Below through the min value at object B, along direction d (cmp. Figure 17(b)).



(a) Plane Central. (b) Planes Above and Below.

Figure 17: Visualized planes which help to outline the acceptance volumes (Fogliaroni and Clementini 2014).

A space where objects (B and C) and their cones are located, can be split in several subspaces. Fogliaroni and Clementini proposed two options:

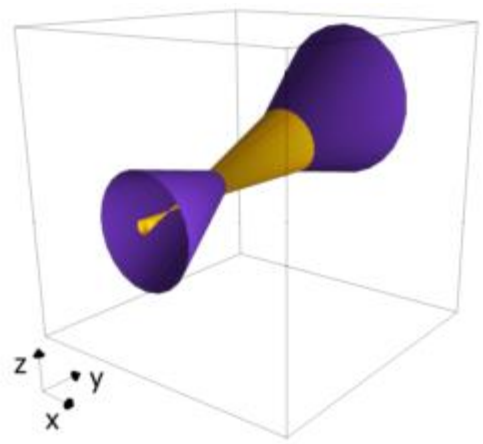
- Relative direction relation model,
- Visibility relation model (coarse visibility model, finer visibility model).

Model for relative direction defines relations, which describe the position of the observed object A in comparison with the reference objects B and C, heading from B to C. The model can distinct between nine volumes (cmp. Figure 18(b)) defined by elements described above (convex hull, planes Central, Above and Below, external and internal cones). Definition of the *acceptance volumes* are presented in Table 1.

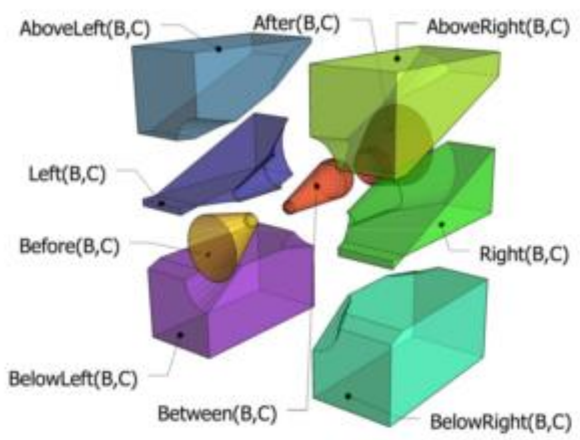
Table 1: Definition of the acceptance volumes in relative direction model (Fogliaroni and Clementini 2014).

Volume	**Definition**	**Corresponding relation**
Between(B,C)	ConvexHull(B∪C)	Between(A,B,C)
Before(B,C)	Δ_{int}^{-} \ConvexHull(B∪C)	Before(A,B,C)
After(B,C)	Δ_{int}^{+} \ConvexHull(B∪C)	After(A,B,C)
AboveLeft(B,C)	$(Central^{-} \cap Above^{-} \cap Below^{+}) \setminus (\Delta_{ext} \cup \Delta_{int})$	AboveLeft(A,B,C)
Left(B,C)	$(Central^{-} \cap Above^{+} \cap Below^{+}) \setminus (\Delta_{ext} \cup \Delta_{int})$	Left(A,B,C)
BelowLeft(B,C)	$(Central^{-} \cap Above^{+} \cap Below^{-}) \setminus (\Delta_{ext} \cup \Delta_{int})$	BelowLeft(A,B,C)
AboveRight(B,C)	$(Central^{+} \cap Above^{-} \cap Below^{+}) \setminus (\Delta_{ext} \cup \Delta_{int})$	AboveRight(A,B,C)
Right(B,C)	$(Central^{+} \cap Above^{+} \cap Below^{+}) \setminus (\Delta_{ext} \cup \Delta_{int})$	Right(A,B,C)
BelowRight(B,C)	$(Central^{+} \cap Above^{+} \cap Below^{-}) \setminus (\Delta_{ext} \cup \Delta_{int})$	BelowRight(A,B,C)

From now on, if an object A is located somewhere in the space (and not intersecting with objects B and C), it is possible to describe it´s relative position in form of R_{dir}(A,B,C), where R_{dir} can be replaced with any defined relation in table. This is the case only for single-tile relations when an object intersects only one volume. For instance when A falls completely in the volume Between(B,C), the relation Between(A,B,C) is true. But if A extends in more than one volume, we are dealing with multi-tile relations. Multi-tile relations are named after merging the single-tile relations corresponding to the volumes. If A is located in volumes Left(B,C) and Between(B,C), the multi-tile relation Left ∧ Between(A,B,C) is true.



(a) External and internal cones.



(b) Exploded acceptance volumes for relative directions.

Figure 18: Relative directional model is obtained by using external and internal cones, convex hull, planes Central, Above and Below (Fogliaroni and Clementini 2014).

Visibility relation model divides a space in different volumes than relative direction model. It follows the idea presented in Section 2.3, where Fogliaroni et al. (2009) defined acceptance areas. The model predicts visibility relations between observed object A, observer B, when a third object C is perceived as an obstacle.

A *coarse visibility model* distinguishes only between three volumes, which are defined by cones and convex hull. By combining both – external and internal – cones (Figure 18(a)) one can extract: **Light Zone** (LZ), **Shadow Zone** (SZ) and **Twilight Zone** (TZ) (Figure 19(a)). If an object A falls into:

- LZ, the relation is named ***Visible(A, B, C)***, which means any point of observed object is visible to every point of the observer,
- SZ, the relation is named ***Occluded(A, B, C)***, since all points of the observed object are occluded to every point of the observer.

- TZ, the relation is named ***PartiallyVisible(A, B, C)***, because an observed object is visible only to some points of the observer.

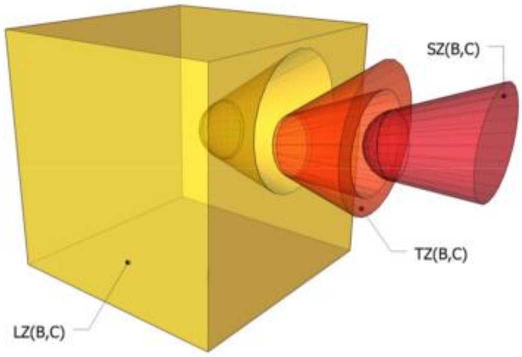
Table 2: Definition of the acceptance volumes in coarse visibility model (Fogliaroni and Clementini 2014).

Name	**Volume**	**Definition**	**Corresponding relation**
Light Zone	LZ(B,C)	$\mathbb{E}^3 \setminus (\Delta^+_{int} \setminus (\text{ConvexHull}(B \cup C) \setminus C))$	Visible(A, B, C)
Shadow Zone	SZ(B,C)	$(\Delta^+_{ext} \setminus \Delta^-_{int}) \setminus (\text{ConvexHull}(B \cup C) \setminus C)$	Occluded(A, B, C)
Twilight Zone	TZ(B,C)	$\Delta^+_{int} \setminus \Delta^+_{ext}$	PartiallyVisible(A, B, C)

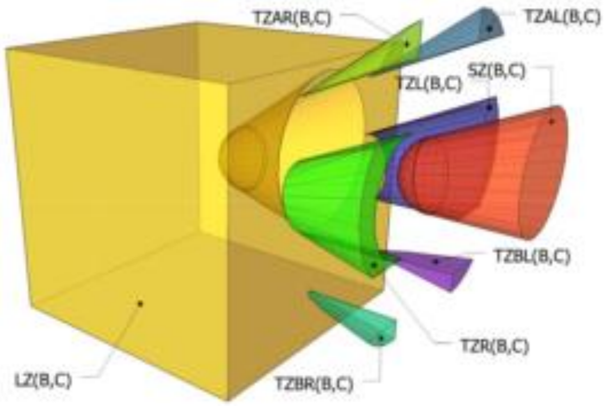
A *finer visibility model* can be obtained by dividing a TZ into smaller volumes by using planes Central, Above and Below, defined above in Figure 17. In such case TZ is divided in subzones: Above Left (AL), Above Right (AR), Left (L), Right (R), Below Left (BL), Below Right (BR) and Joint (J). The latter subzone can be obtained in a case when the external tangent planes intersect each other, therefore a shadow zone would be finite. With respect to the TZ subzone names, the relation names are specified in a similar manner as PartiallyVisible*, where * can be replaced with every subzone. Examples of finer visibility relations of TZ and definition of its acceptance volumes are presented in Table 3.

Table 3: Definition of the subzones of TZ in finer visibility model, based on coarse visibility model and relative direction model (Fogliaroni and Clementini 2014).

Name	**Volume**	**Definition**	**Corresponding relation**
Twilight Zone Left	TZL(B,C)	TZ(B,C)∩Left(B,C)	PartiallyVisibleLeft(A,B,C)
Twilight Zone Right	TZR(B,C)	TZ(B,C)∩Right(B,C)	PartiallyVisibleRight(A,B,C)
Twilight Zone Above Left	TZAL(B,C)	TZ(B,C)∩AboveLeft(B,C)	PartiallyVisibleAboveLeft(A,B,C)
Twilight Zone Above Right	TZAR(B,C)	TZ(B,C)∩AboveRight(B,C)	PartiallyVisibleAboveRight(A,B,C)
Twilight Zone Below Left	TZBL(B,C)	TZ(B,C)∩BelowLeft(B,C)	PartiallyVisibleBelowLeft(A, B, C)
Twilight Zone Below Right	TZBR(B,C)	TZ(B,C)∩BelowRight(B,C)	PartiallyVisibleBelowRight(A,B,C)
Twilight Zone Below Joint	TZJ(B,C)	Δ^-_{ext}	PartiallyVisibleJoint(A,B,C)



(a) Exploded acceptance volumes of a coarse visibility model.



(b) Exploded acceptance volumes of a finer visibility model.

Figure 19: Visibility models (Fogliaroni and Clementini 2014).

2.5.2 Holes and concavities

Fogliaroni and Clementini (2014) have also considered cases when objects have holes and a non-convex shape. The authors first focused on the observed object A. There are only certain possible multi-tile relations that can occur if A has a convex shape. For example, A located in volumes TZR∧SZ∧TZL could be true if A is a convex object, whereas TZR∧TZAR∧TZAL∧TZL could only be possible if A is concave. The only thing that changes, if A has holes or a concave shape, is the extra multi-tile relations. Check Figure 19(b) for visualization.

If reference objects (B, C) have no holes and a concave shape, then Fogliaroni and Clementini suggest that:

1. B should be split into multiple convex parts ($B=B_1 \cup ... \cup B_m$, $C = C_1 \cup ... \cup C_n$).
2. Acceptance volumes must be computed for each pair (B_i, C_j).
3. Volumes must be combined based on aggregation rules.

If objects have holes it influences the visibility relations. Therefore, in this case, Fogliaroni and Clementini suggest separating object holes and real object. Similarly, as before, acceptance volumes need to be calculated for all parts. In case when a part is concave, it must be decomposed into convex parts. Special attention needs to be addressed for holes, because the function of Shadow and Light Zones must be substituted. At the end the volumes should be combined according to aggregation rules.

3 VISIBILITY-BASED NAVIGATION IN 3D

This section describes the visibility-based navigation in 3D space. The process itself is similar to the process in 2D space that was described in Section 2.4. However, the process in 3D space is much more complicated. There are several circumstances that do not exist in 2D, but appear as a challenge in 3D perspective, those will be discussed in the Chapter 4.

In Section 2.5.1 we have presented visibility relation model between observed object A, observer B and obstacle C. Since the purpose of visibility-based navigation in 3D is to successfully navigate people, that visibility model can be generalized and simplified so that the observer object is conceived as a point. Fogliaroni and Clementini (2014) already showed that consequently, when the observer object is a point *b* then the external and internal tangents coincide with one another and thus the PartiallyVisible relations in case R(A, b, C) do not hold anymore. Based on that, we obtain:

$$\begin{gathered}\text{Visible }(A, b, C) \Longleftrightarrow \text{Visible }(b, A, C)\\ \text{Occluded }(A, b, C) \Longleftrightarrow \text{Occluded }(b, A, C)\\ \text{Visible} \wedge \text{Occluded }(A, b, C) \Longleftrightarrow \text{PartiallyVisible* }(b, A, C)\end{gathered} \tag{5}$$

where * could be replaced with Above Left (AL), Above Right (AR), Left (L), Right (R), Below Left (BL), Below Right (BR) and Joint (J).

In Equation (5) are presented relations, where A and C are two reference objects and b is an agent. Originally, when we compute the acceptance volumes (Table 2, Table 3), we calculate them between objects A and C, such that A is in a role of observer and C is an obstacle. Therefore the agent b is the observed object and we deal with R(b, A, C). However, when we navigate an agent trough the environment, the roles are reversed, such that an agent is in a role of observer and A is now the observed object – R(A, b, C).

As in 2D space (Section 2.4), we must pay attention when we want to change the perspective, i.e. transition from R_1(A, b, C) to R_2(b, A, C), because in the case of the Visible ∧ Occluded (A, b, C) the relative direction also changes. Imagine that the observed object is an object A and that only a part of it (A_v) is visible to b. Then the following applies:

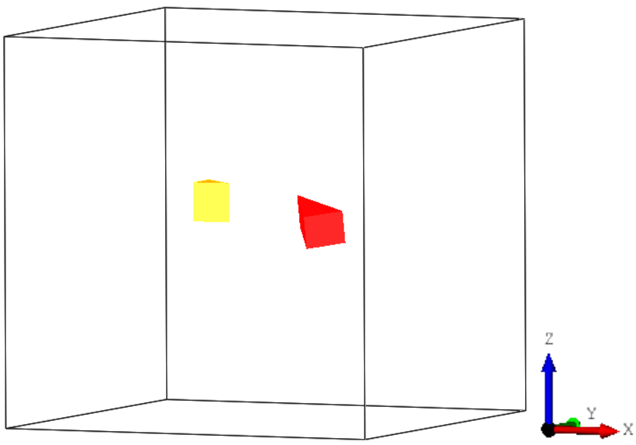
$$\begin{gathered}\text{Visible }(A_v, b, C) \wedge \text{Right }(A_v, b, C) \Longleftrightarrow \text{PartiallyVisibleLeft }(b, A, C)\\ \text{Visible }(A_v, b, C) \wedge \text{Left }(A_v, b, C) \Longleftrightarrow \text{PartiallyVisibleRight }(b, A, C)\\ \text{Visible }(A_v, b, C) \wedge \text{BelowRight }(A_v, b, C) \Longleftrightarrow \text{PartiallyVisibleAboveLeft }(b, A, C)\\ \text{Visible }(A_v, b, C) \wedge \text{BelowLeft }(A_v, b, C) \Longleftrightarrow \text{PartiallyVisibleAboveRight }(b, A, C)\\ \text{Visible }(A_v, b, C) \wedge \text{AboveRight }(A_v, b, C) \Longleftrightarrow \text{PartiallyVisibleBelowLeft }(b, A, C)\\ \text{Visible }(A_v, b, C) \wedge \text{AboveLeft }(A_v, b, C) \Longleftrightarrow \text{PartiallyVisibleBelowRight }(b, A, C)\end{gathered} \tag{6}$$

In Equation (6) were included relations from relative direction model and visibility model, which were defined in Section 2.5.1.

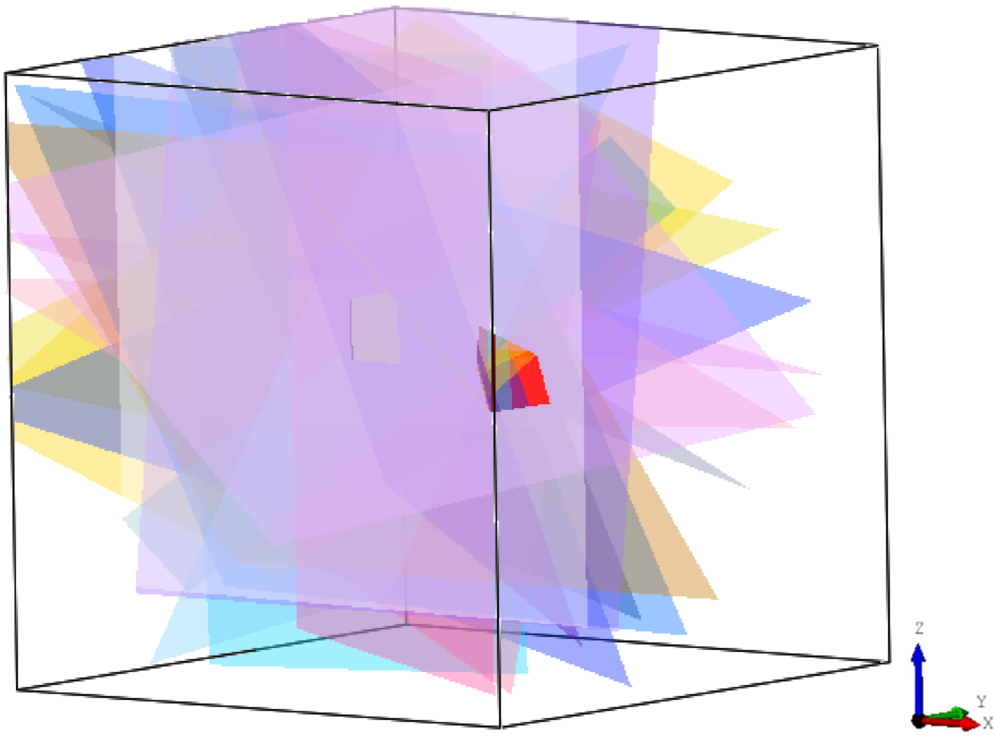
The process of obtaining regions

In order to obtain qualitative coordinates, that provide information of which objects can one see from a certain position in the environment, the space needs to be divided into sub regions. In comparison with 2D, in 3D space regions are volumes instead of areas and borders between them are faces instead of lines. Regions can be obtained by computing external and internal tangent planes (defined in Section 2.5.1) between every pair of objects in the environment.

Imagine a pair of objects B, C (cmp. Figure 20(a)) between we compute tangent planes which are depicted in Figure 20(b).



(a) Objects in 3D.



(b) External and internal planes between objects (a).

Figure 20: External and internal tangent planes computation.

However, we are not interested in the entire plane computed between the objects. Similarly, as in 2D (Figure 10) we want to ignore part of the plane, which is located in volume Between(B, C), instead we are only interested in *half-planes* located in volumes Before(B, C) and After(B,C). Volumes Between(B, C), Before(B, C) and After(B, C) are defined in Section 2.5.1.

Therefore, each plane should undergo a calculation process called *cutting function*:

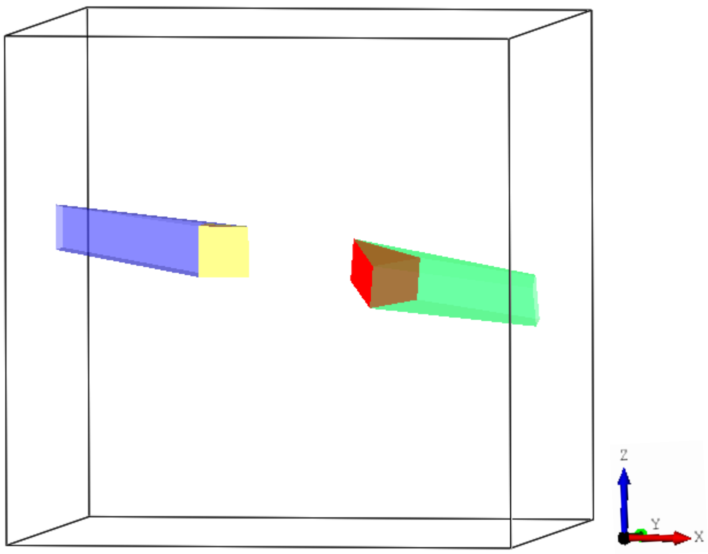
1. Check if plane intersects any other object in volume Between(B, C):
    a. If no: continue to second point
    b. If yes: It could be that this plane should not exist, or only a part of it should exist.
2. Obtain two half-planes, which are located in volumes Before(B, C) and After(B, C)
3. Limit each half-plane by the boundaries of other objects

The mentioned process is similar to the one performed in 2D environment, which is already presented in Section 2.4. But since in 3D environment we are dealing with planes and not lines, the process is more complicated. First restriction is needed because a third object crossing the plane between objects means that all points on the tangent plane will never be observable simultaneously from the points located on a half-planes located in volumes Before(B, C) or After(B, C). Depending on how much is the third object occluding, it is decided if only a part of a tangent plane should exist, or if the tangent plane should not exist at all. This particular restriction will be explained in detail in Section 5.1.1.

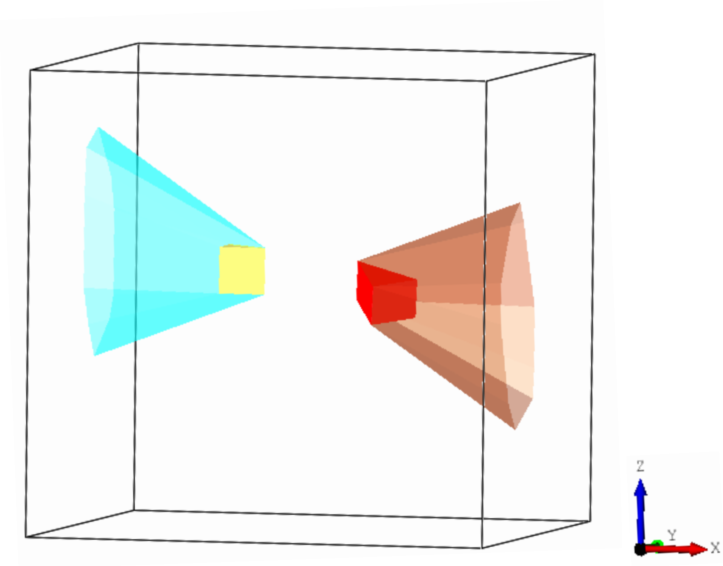
Second step configures two half-planes, by deleting the part of a tangent plane, which is located in the volume Between(B, C).

Third restriction enables that half-planes do not cut through the third object, instead the objects create a hole in them. Again, the process is complicated due to the complexity of 3D environment; therefore this issue is discussed in details in Section 5.1.1.

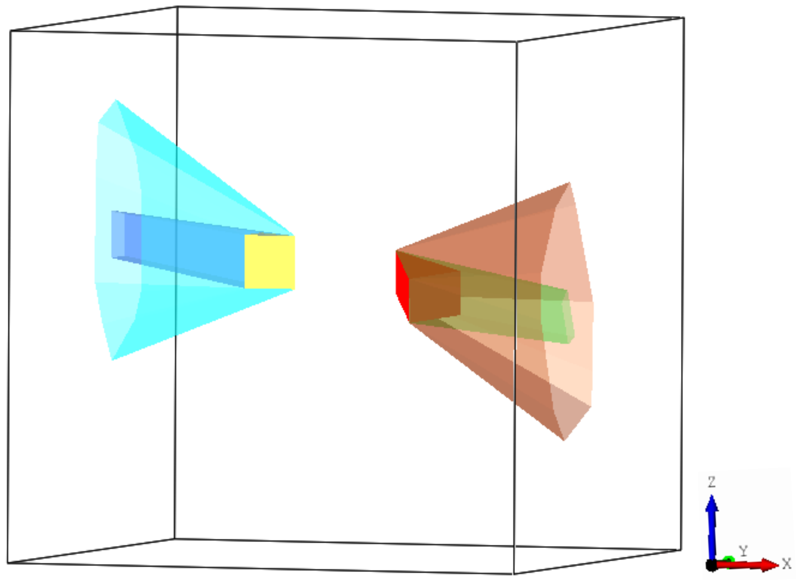
After these steps, we obtain a set of external Π_{ext} and a set of internal Π_{int} half-planes. From which external and internal cones are produced (cmp. Figure 21) as described in Section 2.5.1. With this steps we obtain a coarse visibility model, since we did not use planes Central, Above and Below (explained in Figure 19(a) and Figure 17). Therefore, with cones the following zones are acquired: Light Zone (LZ), Shadow Zone (SZ) and Twilight Zone (TZ). Those zones were in 2D space perceived as areas and boundaries between them were lines. Now in 3D geometry we are engaging with volumes instead of areas and with surfaces or faces instead of lines, where each volume represents a region.



(a) External cones.



(b) Internal cones.



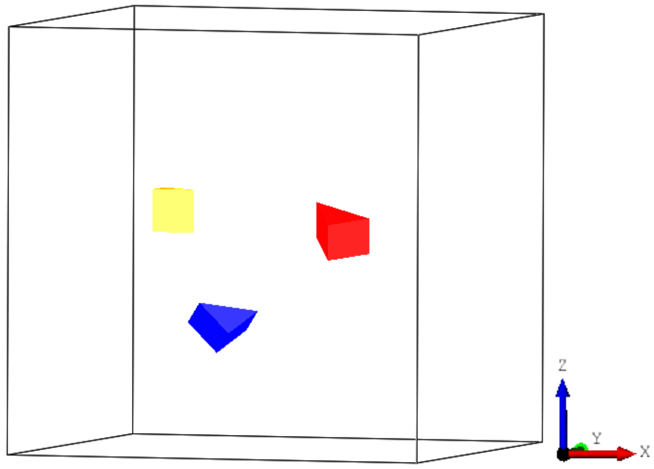
(c) External and internal cones.

Figure 21: Computed external and internal cones between two objects, which divide space into regions.

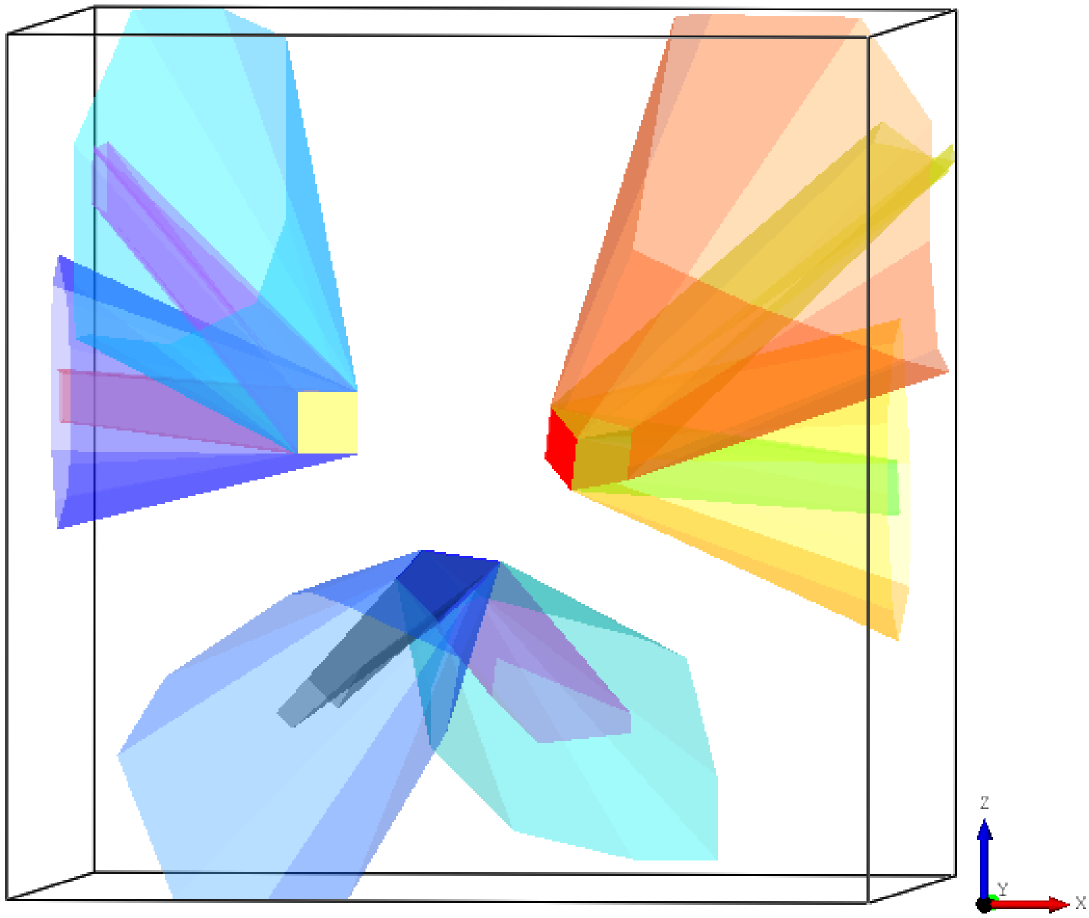
When the regions are extracted, the topological map can be obtained, where:

- Each node represents a region obtained from space subdivision
- Every edge represents the ability to move between regions by crossing only one boundary face.

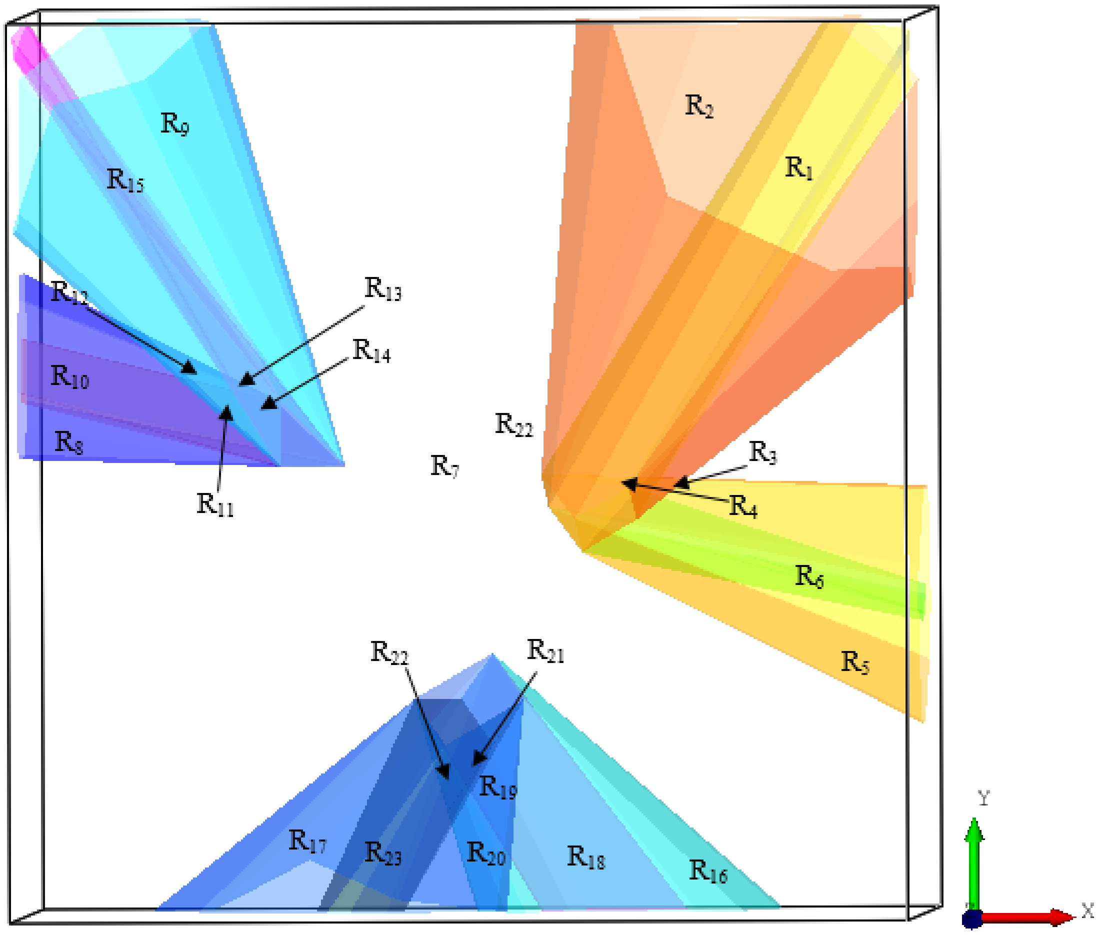
In Figure 22 is an example of space subdivision into regions computed between 3 prisms. The obtained result is based on the process described above, so as in Figure 21 the coarse visibility model is calculated also in Figure 22(b). In the end we obtain 23 regions presented in topological map (cmp. Figure 22(d)).



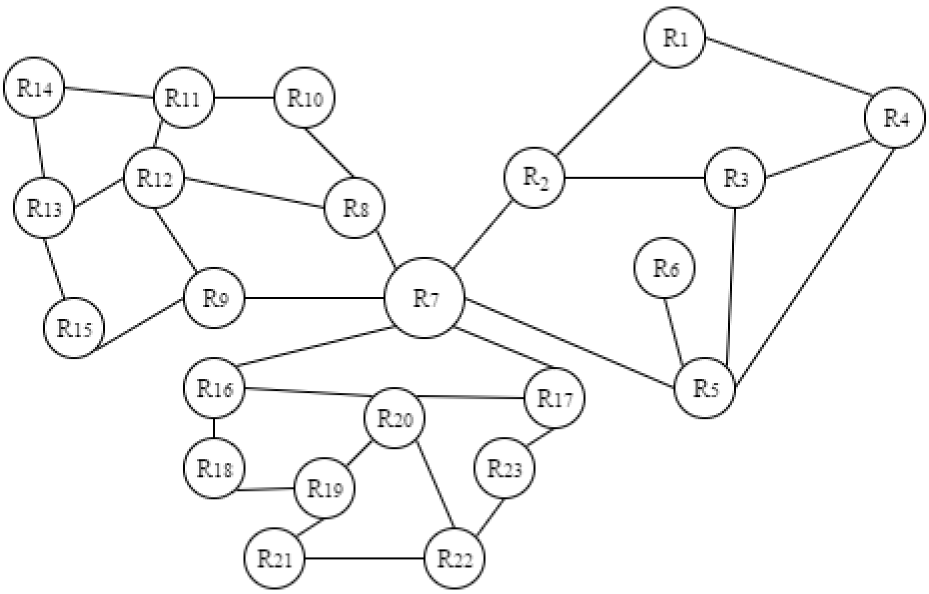
(a) Three prisms.



(b) Space divided into regions.



(c) Space subdivision from different perspective and with marked regions.



(d) Topological map.

Figure 22: Space subdivision in 3D and corresponding topological map based on a coarse visibility model.

The agent needs to be able to differentiate between regions in order to navigate successfully. Each region corresponds to a volume from which it is possible to see certain (parts of) objects, therefore as for the 2D case, in 3D space the acceptance volumes are also characterized on the basis of what the agent can see when he is situated in them. In 2D perspective Visual Cyclic Ordering (VCO) were used for this purpose, which is basically the agent's panoramic view of the environment. In 3D space the perspective of the environment should be described in sphere and not cyclic order. Therefore a Visual Sphere Order should be implemented, which would support all visibility relations in 3D and all multi-tile relations (described in Section 2.5). Topic will be further discussed in Section 5.1.2.

When the VSO will be fully defined a topological map with VSO labels could be produced, similar to the ones in 2D space (Figure 13(b)).

At the end, the aim is to produce landmark-based route instructions, which would cognitively support and navigate the user from the starting point to the end. Those instructions can be: "walk around A keeping it on your right until you start perceiving B on A's left". Landmarks in this case are objects A and B that could be replaced with the real objects from the environment, which have salient characteristics (Sections 2.1.4, 2.2.2), e.g. A could be a desk information centre and B a staircase. Therefore, the main focus of the visibility model is landmarks and the position from which a certain landmark can be seen, so we can refer to them and use them in route instructions. But of course non-salient objects need to be included in the visibility computation as well, since they might occlude landmarks.

3.1 Concrete examples and comparison between them

Computation becomes complex with each new object included in the calculation, since more volumes and more faces are calculated between them. Of course, the final number of faces and volumes vary, since it is sensitive to the layout and the shape of the objects.

To get a better grasp of complexity, we computed a total number of produced faces on four random examples presented in Table 4. The visibility was calculated between the two, three, four or five prisms, for which the produced number of faces was counted. The numbers in Table 4 do not reflect upper-boundaries of possible faces; instead the number is only valid for a certain layout and shape of prisms.

In the case of 2 prisms presented in Figure 21(c) 30 faces are obtained. In Figure 22 with 3 prisms 90 faces were obtained, in Figure 23 with 4 prisms 170 faces were computed, where as in the case of 5 prisms 295 different faces are extracted. The latter case is presented in Figure 24 to visualize the intricacy.

Table 4: Comparison of how many faces are obtained in the case of a certain number of prisms.

Number of prisms	2 prisms	3 prisms	4 prisms	5 prisms
Number of faces	30	90	170	295

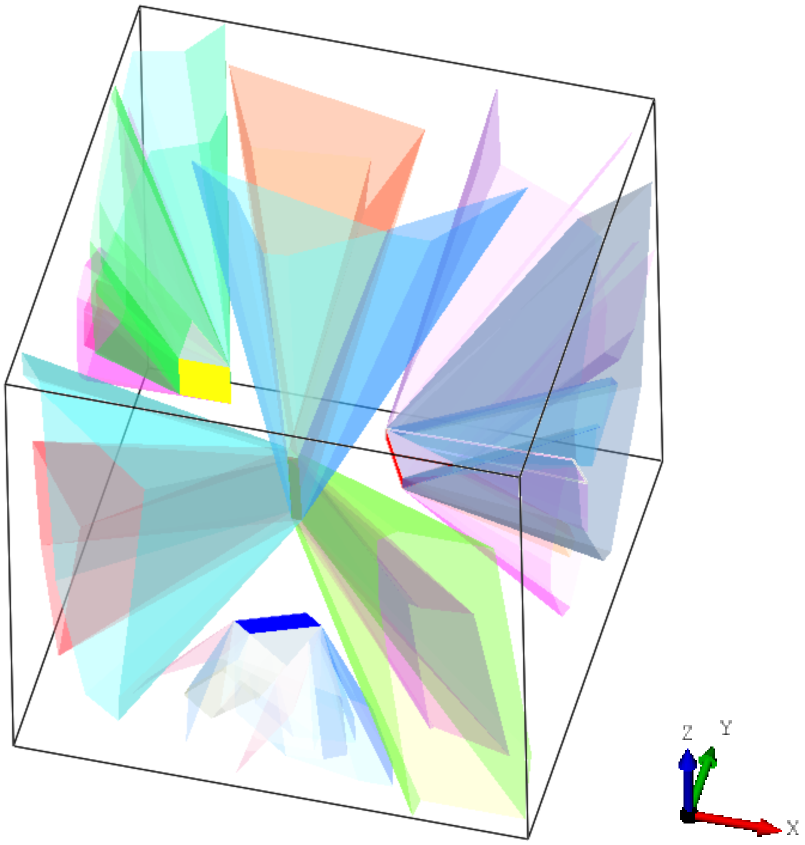


Figure 23: External and internal volumes computed between 4 prisms.

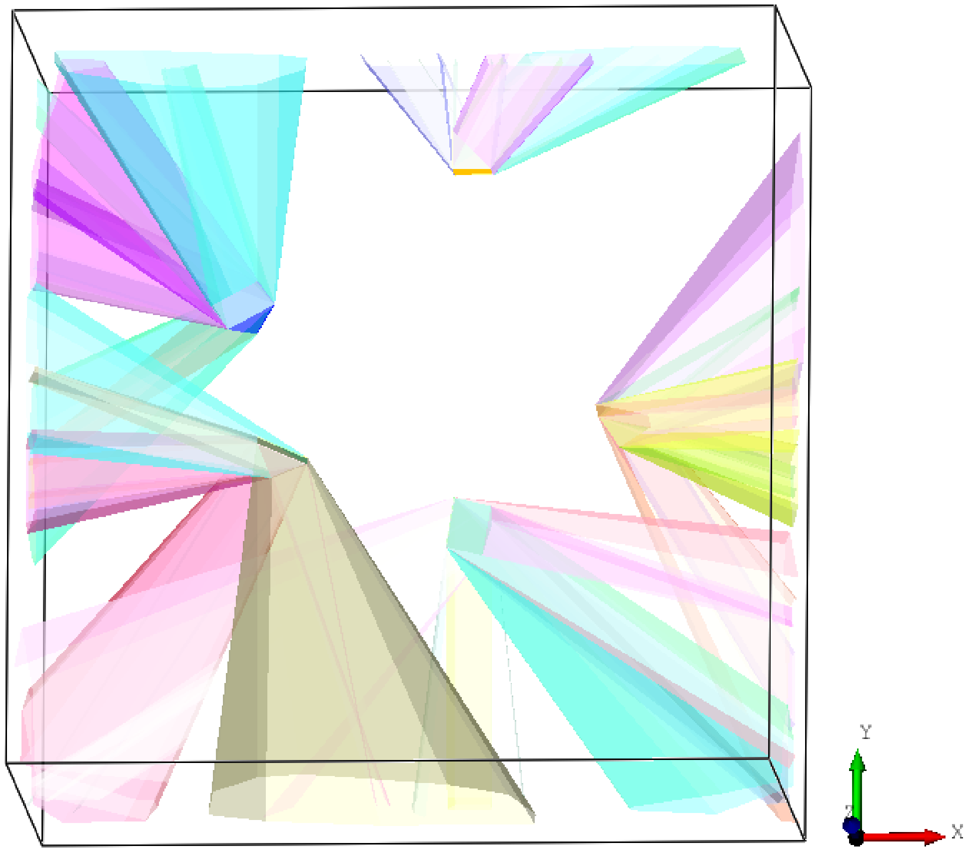


Figure 24: External and internal volumes computed between 5 prisms.

As we already said, the aim is to produce landmark-based instructions, like presented in Example 1. We are not interested to acquire all possible regions in the environment, but only the regions which are describing the visibility of landmarks. We should find a way to simplify it, not just to reduce the computational load, but also to reduce the number of faces and consequently the number of regions.

In the following we will discuss several options that enable to reduce the computational and cognitive complexity of visibility based navigation.

First, we would not consider all objects in the environment in the computation, but instead define the ones that are needed in the production of indoor landmark-based route instructions. Assuming we have a set of objects in the environment O with n elements:

$$O = \{o_1, o_2, o_3, o_4, o_5, \dots, o_n\} \tag{7}$$

Based on some conditions, which will be discussed in Section 4.2, we could exclude some elements from the set O, before computing visibility relations between them:

$$O = \{o_1, o_2, o_3, \cancel{o_4, o_5}, \dots, o_n\} \rightarrow O = \{o_1, o_2, o_3, \dots, o_m\}$$

Therefore, O would contain only m elements where $m < n$.

Second, as it was discussed in this chapter the visibility regions in the space are obtained by applying visibility reference frame between all pairs of objects, where pairs are type of (observer, obstacle).

Which means the object in first position is considered as an observer and object in the second position is seen as an obstacle. This was already discussed in Section 2.5.1 and at the beginning of Chapter 3. Therefore, to obtain the volume shaped regions, we create new set consisting of pairs:

$$O^2 = O \times O = \left\{ \begin{matrix} (o_1, o_1), (o_1, o_2), (o_1, o_3), \dots, (o_1, o_m), \\ (o_2, o_1), (o_2, o_2), (o_2, o_3), \dots, (o_2, o_m), \\ (o_3, o_1), (o_3, o_2), (o_3, o_3), \dots, (o_3, o_m), \\ \vdots \\ (o_m, o_1), (o_m, o_2), (o_m, o_3), \dots, (o_m, o_m) \end{matrix} \right\} \tag{8}$$

In this case O^2 will represent all possible combinations between the elements in set O. However, we should not consider pairs consisting of the same object:

$$\overline{O} = (o_i, o_i) \ \forall\, i = 1, \dots m \tag{9}$$

The set that we use at the end for computing the visibility regions between objects is:

$$\ddot{O} = O^2 - \overline{O} = \left\{ \begin{matrix} (o_1, o_2), (o_1, o_3), \dots, (o_1, o_m), \\ (o_2, o_1), (o_2, o_3), \dots, (o_2, o_m), \\ (o_3, o_1), (o_3, o_2), \dots, (o_3, o_m), \\ \vdots \\ (o_m, o_1), (o_m, o_2), \dots, (o_m, o_{m-1}) \end{matrix} \right\} \tag{10}$$

In order to reduce the number of produced regions and obtain only regions of our interest, some pairs in the set $\ddot{O}$ could be ignored. Discussion which pairs exactly can be ignored will be presented in Section 4.2.2.

Moreover, one could apply several measures to simplify indoor visibility model. These issues will be discussed in Chapter 4 and some will be left for future work (Section 5.1).

4 COGNITIVE AND COMPLEXITY RELATED ISSUES

Navigation based on visibility model in 2D environment was widely covered by Fogliaroni (Section 2.4). However, when navigating in 3D environment, there are new challenges that need to be resolved in order to get the implementation of 3D indoor landmark-based route instructions from the visibility model. The challenges arise due to the complexity of 3D space. As it was already discussed in Chapter 3 we are dealing with volumes instead of areas and planes instead of lines.

Therefore, in this chapter we will address these challenges and point some simplifications which could be applied. We first need to know what the obstacles and landmarks are (Section 4.1). As it was discussed in Section 3.1, we do not want to compute visibility between every object in set O from Equation (7), nor we want to include all elements of set $\ddot{O}$ from Equation (10). Instead a preselection is suggested in Section 4.2. Since indoor environment has objects with special characteristics, we address them separately in Section 4.3. It is important to consider the fact that some objects can be moved and will not constantly exist or be in the same position, for this reason temporal dimension of objects is discussed in Section 4.4. A user of the landmark-based route instructions will always observe landmarks at the height of his eyes, therefore his point of view will be looked into in more detail in Section 4.5. Other simplifications of regions could also be applied (Section 4.6). We propose a combination of metric map and topological graph in Section 4.7.

4.1 What is an obstacle and what is a landmark?

In Chapter 3 we presented that for production of landmark-based route instructions (Example 1) in visibility model the main focus are landmarks, but of course non-salient objects – obstacles – need to be included in the visibility computation as well, since they might occlude landmarks. This means we do not need to consider all objects from the environment in the calculation, but only the one which are defined as landmarks and obstacles. Therefore, we will in the following define what the obstacles are and what landmarks in the computation of visibility regions are.

Landmarks are those objects that have to be referred to in the route instructions i.e. the objects we are searching when navigating. Their characteristics were already explained in Section 2.1.4, where it is emphasized that landmarks are objects with high salience and described as the most common objects inside the buildings that people consider to be landmarks (Section 2.2.2). Assume we have a set of object in environment O, already defined in Equation (7) then a set of landmarks O_L is defined as:

$$O_L \subseteq O$$

Obstacles are all objects, which we do not consider as landmarks, i.e. do not have characteristics of objects which would stand out from the environment, but are occluding the landmarks because of their position or size and therefore need to be included in the computation of visibility. Set of obstacles O_O is defined as:

$$O_O \subseteq O$$

Note that O_L and O_O are not necessarily to be disjoint, i.e. a landmark could be also an obstacle and vice versa.

4.2 Object preselection

Depending on the discussion in Section 4.1 before computing visibility regions one needs to preselect object such as landmarks and obstacles. Such action will reduce the number of unnecessarily computed regions, since the number of objects, which are included in the visibility calculations, directly influence the final amount of regions (Section 3.1).

Following the Equation (7), which represents a set of all objects in the environment, we need to define which of those are a member of O_L, and which are part of O_O:

$$\begin{gathered} O = \{o_1, o_2, o_3, o_4, o_5, \cancel{o_6, o_7}, \dots, o_n\} \\ O_L = \{o_1, o_3, o_5, \dots, o_n\} \\ O_O = \{o_1, o_2, o_4, \dots, o_n\} \end{gathered} \qquad (11)$$

This will be our **filter**, which will eliminate objects from set O because we are not interested in them. Based on the example in Equation (11), with this filter we will exclude objects o_6 and o_7, because they are not a member of O_L nor O_O. Consequently, we will not produce any visibility regions connected to excluded objects, since they will be eliminated from the computation. The new set O has therefore m elements (m < n). Specifications about the first filter will be discussed in Section 4.2.1.

In section 4.2.2 we will discuss in more details what the consequences of filtering are.

4.2.1 Filter

Depending on the discussion above in first filter we will select landmarks (O_L) and obstacles (O_O) from set O.

4.2.1.1 Selection of landmarks

Since landmarks are our main interest when producing the instructions, it is recommended to initially focus on extracting them. First, saliency of every object in the environment needs to be calculated (discussed in Section 2.1.4) to get the objects which could potentially be used in the instructions. Out of all the possibilities, the objects that have a low saliency value can be eliminated, i.e. have lower value of the surrounding area. So if there are many doors, they cannot be considered as landmarks because there are too many similar objects in the immediate vicinity. They are therefore not unique enough and do not stand out from the surrounding area as a single object, but as a group (unless they have specific visual properties as different colour, shape). Despite the fact that doors fall into the group of functional objects, which according to the Ohm et al. (2014) are considered the most referred objects in indoor environment by people (Section 2.2.2), in this case they still cannot be used as landmarks.

The models presented in Section 2.1.4 are already a basis for the computation of saliency. Therefore, landmarks can be possibly selected based on models of Raubal and Winter (2002), Klippel and Winter (2005) or Duckham et. Al. (2010).

4.2.1.2 Selection of obstacles

As already presented in Section 4.1, obstacles are objects that are occluding landmarks, which means they are heavily dependent on landmarks and their properties. Decision, which object really is an obstacle, is dependent from the viewpoint of the agent and the relative size of the projected obstacle with respect to the size of the projected landmark. Therefore the relative size is more relevant than the absolute size of the objects. However, this projection cannot be obtained that easily, since the viewpoint of the agent changes. Moreover, some objects in the environment are not always in the same position or their orientation changes. For that reason, we will discuss other options in the following that can be applied instead.

When trying to define or select obstacles in the environment it should be assumed that every element of set O is an obstacle, until it is proved otherwise:

$$O_O = O \tag{12}$$

The aim is to apply criteria which would eliminate some elements from set O_O, after it is proven they are not occluding any landmarks. We propose the usage of the following criteria:

1. Position of the object
2. Dimensions of objects in x, y, z directions
3. Semantic

First two restrictions are purely geometrical based, whereas the third restriction focuses on semantic information. In the following we will analyse each criterion, how and when should they be applied and used.

1. Position of the object

Each object in the environment has a point P, which defines a position of an object i.e. its location in space:

P (x, y, z)

Point P, defined by coordinates, can be used as a criterion that describes which objects in visibility model are and which are not obstacles. To provide good simplifications, we need to remember that we are searching for objects which are from agent's point of view visually occluding other object (landmarks). So we need to focus on two different positions:

- position of an agent
- position of landmarks

The position of an agent, while navigating in 3D environment, is normally changing in xy-plane. However, his position is more or less fixed in z axis, since he observes the environment through his eyes, which are located on a certain height - *h*. So we could assume that agent will never be located higher than his height. However, the aim of 3D navigation based on visibility is to navigate people of different height. We need to consider to navigate a person in a wheelchair (ca. 1.20 m) and the tallest living man, whose height is around 2.50 m[3].

The position of landmarks differs from each other, so it is necessary to find a solution that would suit all of them at the same time. For that reason, we propose to define a minimum bounding box for each floor which would contain all landmarks.

$$bounding\ box = \ x_{min}^{L}, y_{min}^{L}, z_{min}^{L}, x_{max}^{L}, y_{max}^{L}, z_{max}^{L}$$

From Table 5 it is visible that it is possible to obtain the minimum and maximum z value for the agent as for the landmarks. However, this is not possible for the values in x-axis and y-axis. In other words, we can create some criteria to define objects which are not occluding landmarks based on the z-axis but not on the x and y axis.

Table 5: The comparison of possible positions of an agent and landmarks.

	X axis	**Y axis**	**Z axis**
Position of an agent	freely	freely	*1.20 m – 2.50 m*
Position of landmarks	$x_{min}^{L} - x_{max}^{L}$	$y_{min}^{L} - y_{max}^{L}$	$z_{min}^{L} - z_{max}^{L}$

We propose that for every element in set O_O a minimum bounding box or at least z_{min}^{O} and z_{max}^{O} value are calculated and later on compared with values z_{min}^{L} (resp. z_{max}^{L}) and agent's height, as written in Equation (13). If the equation is true then the element can be eliminated from set O_O.

[3] http://www.guinnessworldrecords.com/world-records/tallest-man-living

$$\max(2.50, z_{max}^{L}) < z_{min}^{O} \quad or \quad \min(1.20, z_{min}^{L}) > z_{max}^{O} \qquad (13)$$

Visual presentation of Equation (13) is summarized in Figure 25. Minimum bounding box, which contains all landmarks (objects in black colour), is presented with green colour, whereas users height is somewhere in between z_{min}^{L} (resp. z_{max}^{L}). Based on the proposition, all objects which are located lower (resp. higher) than z_{min}^{L} (resp. z_{max}^{L}) (red colour) will be eliminated from O_O.

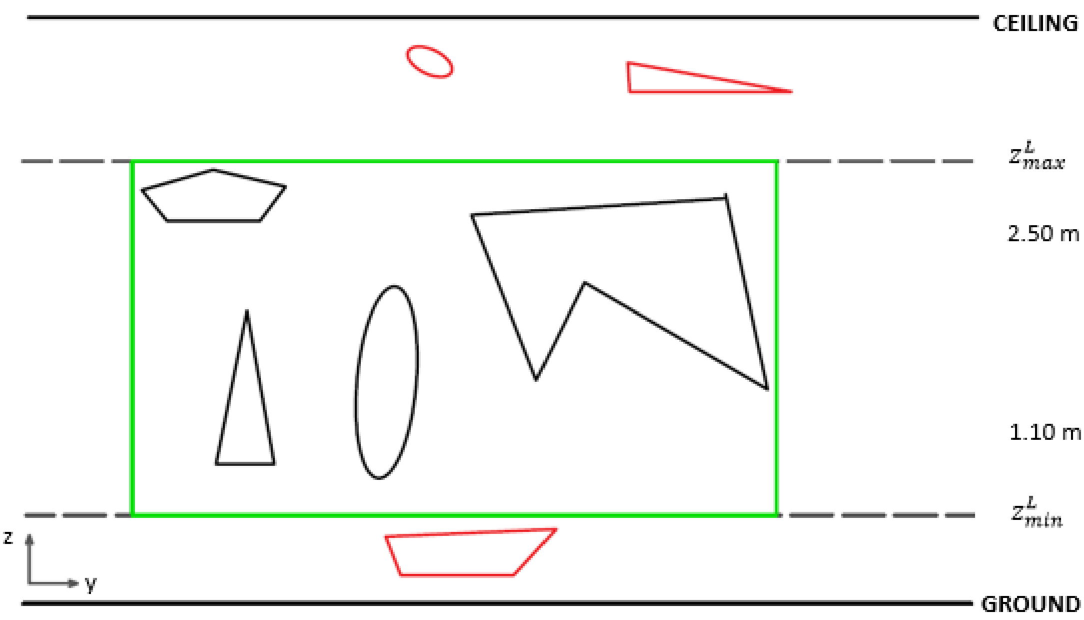


Figure 25: Visual presentation of the Equation (13).

As already mentioned Equation (13) could be applied to (i) each floor separately or to the (ii) whole building at once, but in the latter case one should normalize z values of all objects. It is expected that the (ii) approach eliminates less objects than the (i).

It is important to realise that the Equation (13) can be applied only in a building where each floor is completely exclusive from the other floors and one is not able to see from one floor to another. An example of this type of building is presented in Section 4.3.3. So before utilizing the Equation (13), every floor should have a ground and a ceiling, like in Figure 25.

2. Dimensions of objects in x, y z directions

For now, we have discussed about elimination by only relaying on the position of objects, whereas now we will focus on their dimensions in x, y, z directions, i.e. length, width and height. The aim of this step is to eliminate objects which do not have an impact on the visibility of landmarks because of their small size. With this we have especially in mind the simple standard elements, which are commonly found in buildings like fire sprinkler heads, fire detectors, cameras, emergency lighting etc.

We are aware every object has its own shape, which may be complex. To simplify its complexity, we will only consider dimensions of object's bounding box (x^O, y^O, z^O) for every element in set O_O separately:

$$bounding\ box = \ x^O_{min}, y^O_{min}, z^O_{min}, x^O_{max}, y^O_{max}, z^O_{max}$$

$$x^O, y^O, z^O = (x^O_{max} - x^O_{min}, y^O_{max} - y^O_{min}, z^O_{max} - z^O_{min})$$

With dimensions x^O, y^O, z^O it is possible to analyze suitable thresholds, which would help to distinguish between obstacles and non-obstacles.

In this paragraph we will discuss *volume* and its suitability for a threshold. We know that objects with large volume occupy a lot of space whereas objects with small volume occupy small amount of space. By first instinct one could say that small volumes define small objects and that we should for this reason use volume as a criterion to find small objects that can be eliminated from set O_O.

Well this instinct is wrong. In the dataset, an object with a small volume size, but still occluding many landmarks, could exist. Those objects could be for instance a poster (Figure 26), a flag, a curtain, etc. Their volume size could easily not meet the volume threshold restrictions, but in spite of it they should still be included in the computation, since they could block the view. For that reason, thresholds relying on volumes are not suitable criteria for dividing obstacles and non-obstacles.

Figure 26: This poster should be included in visibility calculation (Poster 2016).

As we have already argued at the beginning of this section, the correct approach would be to consider the relative size between obstacle and landmark. Therefore, to obtain or remove the right obstacles, one should consider the relative distance, position and size of obstacles with respect to the landmarks. Due to the complexity of obtaining the relative values and because the objects are not fixed in space, we propose to consider objects *absolute values* (x^O, y^O, z^O) in order to detect small objects. In sense that that the biggest dimension of an object should not exceed a certain threshold t:

$$max(x^O, y^O, z^O) < t \qquad (14)$$

The only ambiguity that remains is, what value should be applied for t. Decision will be different for each building i.e. each database. Some databases might include all details of the environment, whereas other might not even contain small objects like electrical switch, smoke detectors, etc. We propose to perform several tests with different thresholds and compare the results with each other. Start with the value t = 10 cm and store all eliminated objects in the set O_O^{elim1}, which should be later on analysed. If O_O^{elim1} contains any unexpected objects then t should be reduced, otherwise it can be increased. Afterwards a testing should be repeated until the appropriate quality of results is reached.

The goal of these criteria is only to avoid the objects which have no impact on our outcome of the visibility region computation. The assumption is that the landmarks are large, quickly noticeable and standing out from the environment as it was already mentioned in Section 2.1.4, therefore small objects, like emergency lightning etc. should not occlude them significantly. It may happen that the object by itself could be eliminated from set O_O and would not substantially occlude the view, but if there are numerous objects of similar size in its immediate surroundings, they as a whole should be included in computation. This case is presented in Figure 27, where Equation (14) should not eliminate small objects like this.

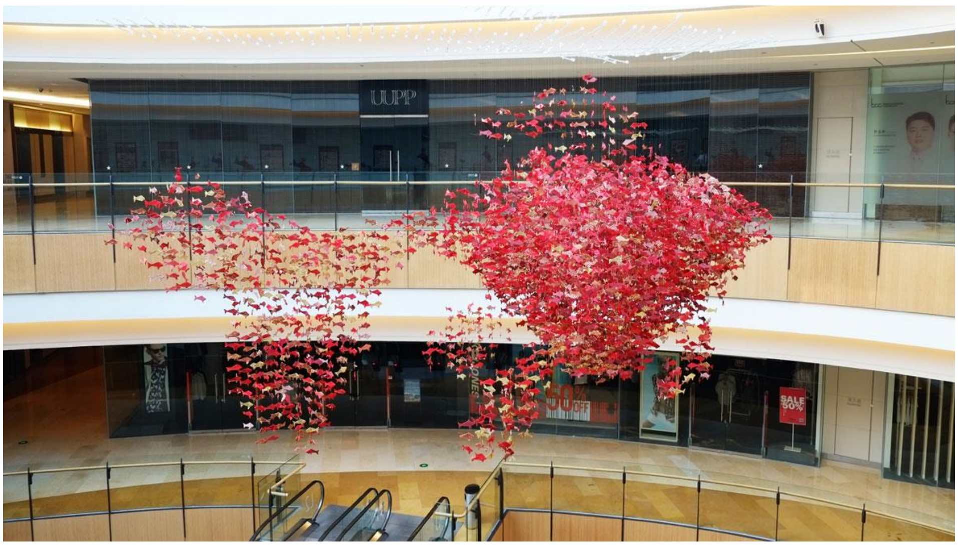
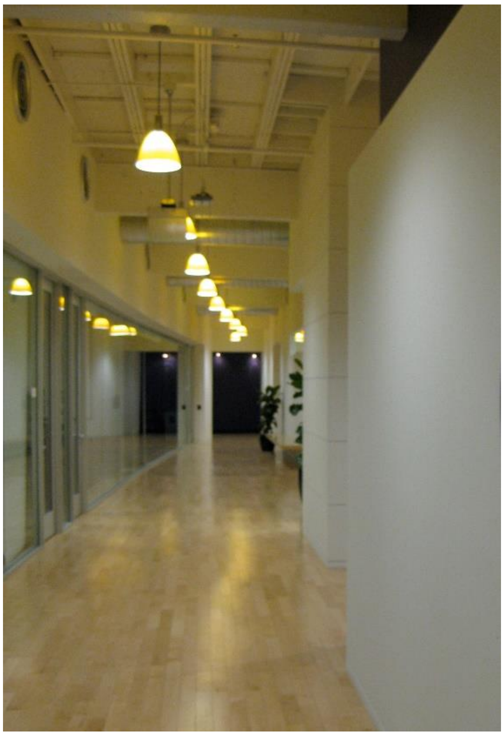


Figure 27: A single paper object will not necessarily occlude the agent's view, unless when they are dense together (Ornament 2016).

3. Semantic

For now we have eliminated objects from set O_O based on their geometrical properties. It is also possible to do so basing on their semantic information, though this is highly dependent on the dataset of the indoor environment. Assuming we have a database which enables the storage of objects in several semantic classes like walls, windows, doors, lamps, alarms, etc. It would be convenient to simply eliminate some classes from set O_O, which are not occluding landmarks, because of their size, position in the environment or their function. These classes might be: electrical infrastructure (lamps, alarms, etc.) fire safety equipment (fire detectors, fire extinguishers), etc. However, before removing a class from set O_O it should be verified that no member of a class is occluding landmarks or perhaps even is a landmark. For example, we will focus on lamps. If we are interested in computing visibility regions in the office-like building, lamps are normally situated at the top, with no special outstanding characteristics like presented in Figure 28(a). In that case lamps could be removed from set O_O. In contrary lamps in different building like in Figure 28(b) should be included in set O_O, since they could occlude some objects.



(a) Lamps which could be removed from set O_O (Lamps1 2016).

(b) Lamps which couldn't be removed from set O_O (Lamps2 2016).

Figure 28: Object elimination based on semantic information should be always carefully executed.

To summarize criteria of elimination are not only strongly dependent on the landmarks but databases as well. We recommend always applying first criterion. Ideally it would substantially reduce number of elements in set O_O. But it may occur that this criterion would not eliminate any elements from the set

O_O, because z_{min}^L and z_{max}^L would not be different from z_{min} and z_{max} of the floor. Second and third criteria are quite similar, because both detect small objects. If the dataset is supported with the semantic information, we propose the usage of the third criterion instead of or before the second, because it enables more control during elimination process as the second criterion.

4.2.2 Consequences of filtering

In Sections 4.2.1.1 and 4.2.1.2 we have discussed a preselection of set O_L and set O_O to reduce the number of elements in set O (defined in Equation (7)) and consequently the number of computed visibility regions. As presented in Equation (10) the set O is used for creation of set $\ddot{O}$, which represents pairs of objects used for the computation of visibility regions. Rather than considering all pair of objects from set $\ddot{O}$, only a subset of it will be used.

We know the pairs in set $\ddot{O}$ are type of (observer, obstacle). Assume we have an object C, such that:

$$C \in O_O \wedge C \notin O_L$$

We can ignore all pairs where C is in the role of observer:

$$\ddot{O} - \{(C, o_i) \mid o_i \in O_O\} \tag{15}$$

For an example if we look at objects from the Equation (11), we can see that element o_2 is not a landmark, but it is in the set O_O. In that case o_2 should only be in a role of obstacle. Therefore, we can eliminate pairs:

$$o_2 \in O_O, \quad o_2 \notin O_L \qquad \ddot{O} = \left\{ \begin{matrix} (o_1, o_2), (o_1, o_3), \dots, (o_1, o_m), \\ \cancel{(o_2, o_1)}, \cancel{(o_2, o_3)}, \dots, \cancel{(o_2, o_m)}, \\ (o_3, o_1), (o_3, o_2), \dots, (o_3, o_m), \\ \vdots \\ (o_m, o_1), (o_m, o_2), \dots, (o_m, o_{m-1}) \end{matrix} \right\} \tag{16}$$

Furthermore, assume there is an object A, such that:

$$A \notin O_O \wedge A \in O_L$$

We can ignore all pairs where A is in the role of obstacle:

$$\ddot{O} - \{(o_i, A) \mid o_i \in O_L\} \tag{17}$$

For instance if we take an example from the Equation (11), there is an element o_3, which is defined as a landmark but not as an obstacle. If we utilise Equation (17), one is able to obtain:

$$o_3 \notin O_O, \qquad \ddot{O} = \begin{Bmatrix} (o_1, o_2), \cancel{(o_1, o_3)}, \dots, (o_1, o_m), \\ (o_2, o_1), \cancel{(o_2, o_3)}, \dots, (o_2, o_m), \\ (o_3, o_1), (o_3, o_2), \dots, (o_3, o_m), \\ \vdots \\ (o_m, o_1), (o_m, o_2), \dots, (o_m, o_{m-1}) \end{Bmatrix} \qquad (18)$$
$$o_3 \in O_L$$

Let us discuss what does the Equations (15) and (17) mean in practice. Assume we have two objects:

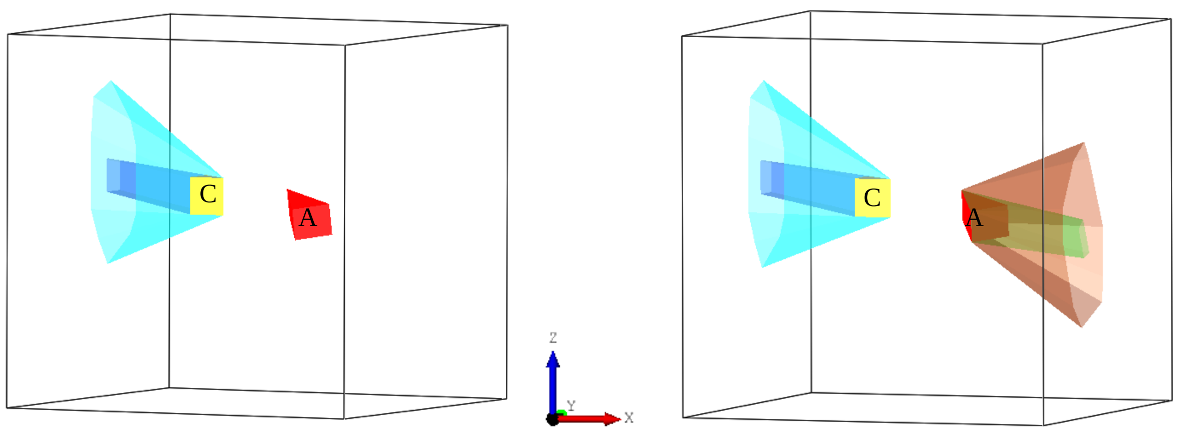
$$C \in O_O \land C \notin O_L \qquad \land \qquad A \notin O_O \land A \in O_L$$

We should only consider a pair (A, C) and not (C, A), when computing visibility regions between objects A and C. Such way, we would produce regions behind the object C and no regions behind the object A like in Figure 29(a). The blue regions (behind C) are defining the positions where the object A (landmark) is only partly visible or occluded. However, if we would produce regions behind A, they would describe visibility of object C which is only an obstacle and has no salient properties. Therefore, we want to avoid the production of those regions, because we are not interested in them.

If we have objects, where:

$$C \in O_O \land C \in O_L \qquad \land \qquad A \in O_L \land A \in O_L$$

We should include both pairs (A, C) and (C, A). Therefore visibility regions will be produced behind A and C, because they are both landmarks. We want to describe visibility and occlusion for both of them. Figure 29(b) is corresponding this situation.



(a) Visibility regions when C is an obstacle and A is a landmark.

(b) Visibility regions when C and A are landmarks.

Figure 29: Visibility regions in different scenarios.

4.3 Special indoor elements

In the indoor environment there are special elements which have a prominent, unique role and because of that they offer opportunities for simplification.

4.3.1 Staircases

Staircases by itself are landmarks and a navigable space, where one can walk through. So they could be used in the instructions like: (i) “pass the staircase” or even (ii) “use the staircases and go to the second floor”. In the second example beside the staircase no additional landmark was used that would describe the path to the second floor. The staircase itself guides the user in the desired direction therefore there is no need to look for landmarks or obstacles. All objects, which are located in the staircase, may be excluded from the computation of visibility. The only parts of the staircase, which are important for visibility are therefore every new beginning or end of the floor, which are in Figure 30 marked with the letters A and B. In case of the indoor database supporting staircases as a type of indoor objects, these restrictions can be used. However, one should be cautious and distinct between corridor-like staircases that lead to a new space and just a couple of stairs. The latter should not be excluded.

Of course, this would influence on the production of the topological map. Letters A and B in Figure 30 denote different visibility regions, therefore they are seen as nodes in a topological map. Further, a staircase between them represents an edge.

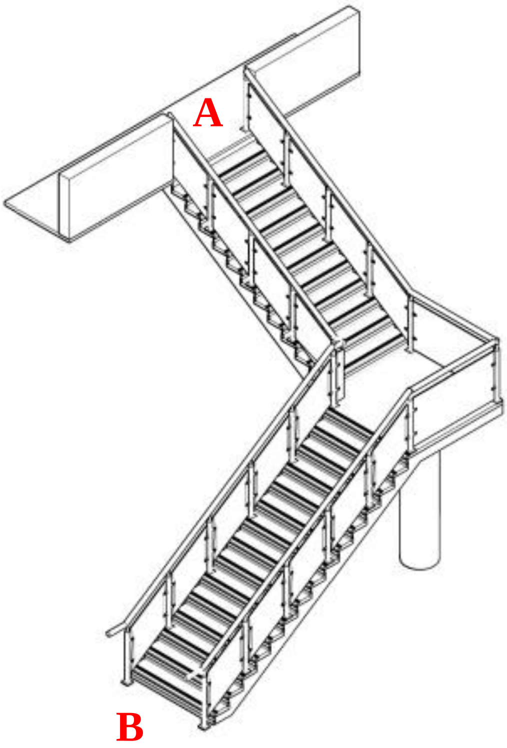


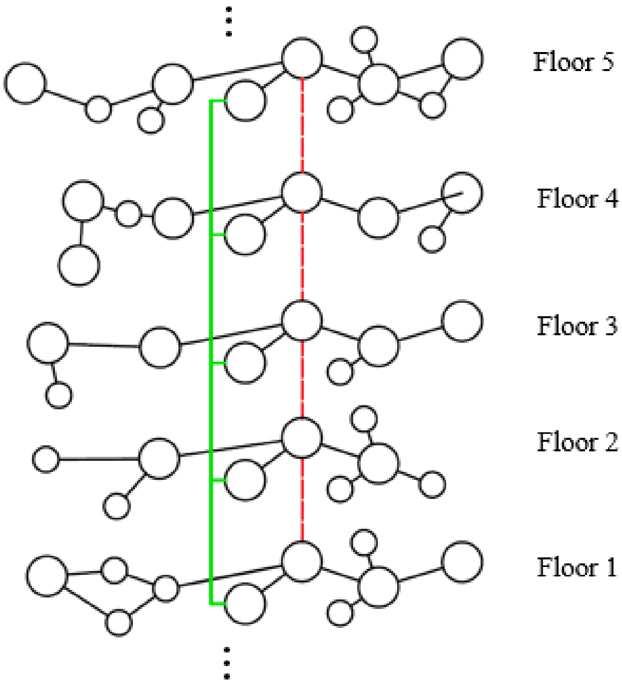
Figure 30: Staircase example, where visibility could be calculated only at levels A and B (Staircase 2016).

4.3.2 Elevators

The elevator has a similar role as staircase in the indoor environment. It is a very particular structural and functional element, which symbolizes a shortcut within our graph, because it allows us to connect two nodes in the graph, which are otherwise not connected. In case of the staircase, one could compute visibility if preferred, but we choose not to in order to reduce regions and computation. In case of an elevator, this cannot be done, since it is a closed element, which shifts the user to the desired location and therefore a visibility computation is not needed.

Assuming we have a building as depicted in Figure 31(a) and our goal is to produce topological map for that building as discussed in Chapter 3. Based on the simplifications proposed for staircases and elevators an approximation of topological map is presented in Figure 31(b). Black circles represent generalized topological regions i.e. volumes. Connection lines between each floor represent the possibility to move from one floor to another. Red hatched line symbolizes the staircase, whereas green line represents the elevator. We distinguish between these lines, because as it was discussed above, the elevator connects the regions which are normally not connected, so one could traverse from first to fifth floor directly without passing any other visibility region. In contrary with staircases, when one travels from first floor to fifth floor, one will pass some visibility regions. It is up to us if we include these regions in the route instructions or not.

(a) Cross section of a building (Cross_Section 2016).



(b) An approximation of a corresponding topological map of the building, where green colour represents the elevator and red colour represents the staircase.

Figure 31: Cross section of a building and a corresponding topological map.

4.3.3 Floors

To reduce the computational load and avoid unnecessary computations, we could exploit another measure, which would take the advantage of the characteristics of the indoor space. In Chapter 3 we said the computation of visibility will be performed between every pair of objects. However usually the object situated on one floor is not visible from the other floor. Assume we have a pair of objects where one is located in first floor and the other in the third floor. When computing visibility between them no regions would be obtained, because the ceiling is between them and they are not visible between each other. But we knew that before the computation. Therefore, we could simply skip calculation between objects which are located in different floors.

However, this simplification cannot be performed in any type of a building. Let us define two different types of building. Imagine a building where each floor is enclosed space i.e. one is not able to see from one floor to another. We will call this type of a building *office-like building*. In contrary imagine a building where each floor is not enclosed space and therefore one is able to see from one floor to another (cmp. Figure 32). The latter type of a building will be named *shopping centre*. In office-like building we can apply the restriction that was discussed above. However, this is not possible in shopping centre, because it may happen that object from one floor is visible to the object in another floor (cmp. Figure 32). Therefore, visibility regions should be computed between them.

Figure 32: Indoor environment, where floors are not excluded from each other (Shopping_Mall 2016).

4.3.4 Rooms

Similar logic as for floors could be applied for rooms. By room we mean a space, which is completely bounded by walls and doors. Consequently, object in one room is not visible to the object in the other room, because the wall is between them. Before applying this rule, it is necessary to ignore definition of rooms in the database of the environment and make a new selection based on our own definition of what the room really is. Suppose that we have in database of the environment two rooms - lobby and cafeteria - as in Figure 33. If rooms are not entirely separated by walls and doors, then they should be treated as one room.

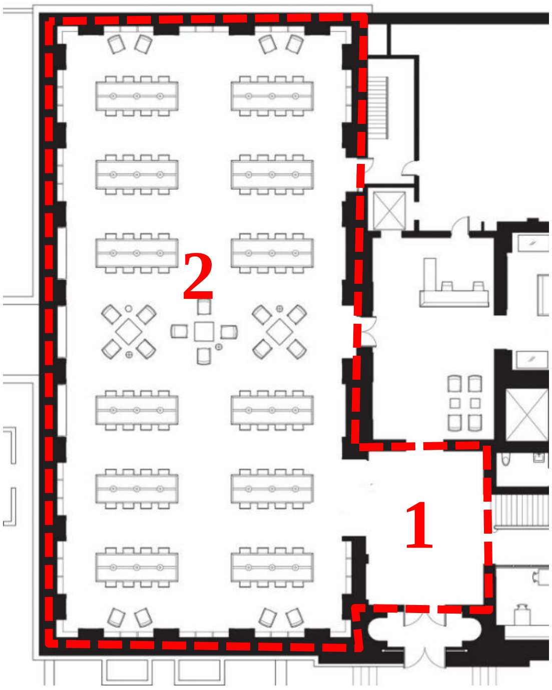


Figure 33: Based on our definition lobby (space 1) and cafeteria (space 2) are considered as 1 room, because there is no door between them (Floor_plan 2016).

We said that if objects are located in two different rooms, they are not visible to one another. This is true if walls are constituted from opaque materials. Objects built out of glass or transparent materials should be treated differently, because they provide a physical obstacle and not a visual obstacle. This topic will be discussed in detail in Section 5.1.4.

4.4 Temporal dimension of objects

In order to successfully use the visibility model for navigation, the data needs to be up to date. With that assumption one can avoid the following situations:

1. using objects that are not present in the environment anymore
2. using objects that are not fixed in space and can be located in different position in the environment that it is recorded in the database
3. intermittent objects
4. the database is completely outdated and does not match with the structure of the building
5. new objects are placed in the environment, which are not stored in the database

The consequences of unsuitable use of the object are certainly different if the object is a landmark or if it is an obstacle. Focusing on the first case (described above), if a missing object would be a landmark, it may happen that the user gets confused by instructions referring to a non-existing landmark or, in the worst case, he gets completely lost and cannot proceed with the navigation. Similar problems arise in the case of a non-existing obstacle. The user would be informed that the landmark is occluded or partially visible, where in fact it would be completely visible. However, in this case, the user is less likely to get completely lost.

In the second case, false usage of landmarks might lead the user to a completely different direction like: “walk around A keeping it on your right”, but if an object is moved the instruction should be: “walk around A keeping it on your left”. Or it might occur that the object is not even located in that room and then the situation is the same as in the first case. But if we would use an unfixed obstacle, two different scenarios could emerge. The first would be that the moved obstacle could in reality occlude a landmark, where in the database it does not occlude it, which is worse than a second case where a landmark could become visible, although it is computed as occluded.

Entities that we have in mind in the third case are the objects which are present only at certain hours of the day or over a longer period of time, but then they disappear. If the database was constructed using the actual state of the environment for the month of December, it should be realized that due to holidays, the appearance of the environment at that time is altered in comparison to the other time period (Figure 34).

Figure 34: Indoor environment has a completely different appearance in December (Holiday_Season 2016).

The scenario described in example 4 is the worst case that can happen in visibility-based navigation. Due to its complete obsolescence of data, the user is guided along a path that does not exist, so navigation in this case is disabled.

In the fifth case it may happen that a new or renovated object has a more salient role than the other elements in the environment. This means that the route instructions do not use the most appropriate landmark. This would have a negative impact on the quality of the navigation however it would not disable it like in some cases described above. The effect could be completely different in the case where a new object is an obstacle. Because the newly appeared object might impair the view of landmarks, or in the best case would not affect navigation. So these are the different scenarios that may arise in the event of our database not being consistent with the actual situation in the environment.

Some situations, like cases 1, 4 and 5 discussed above, cannot be avoided unless the database is updated regularly or is updated when major changes happen.

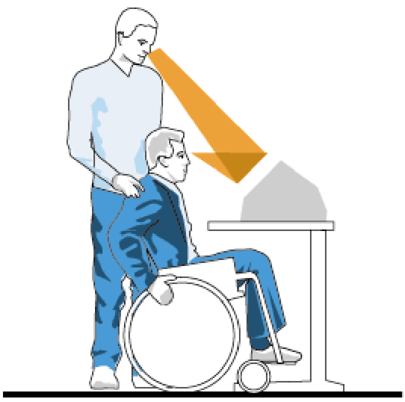
While the circumstances in cases 2 and 3 when we have a landmark may be prevented in such way that the objects, which can be moved or are not visible the entire time, would not be selected as the most salient objects. Instead, they should be assigned a lower value, when calculating the adequacy of the objects as landmarks (Equation (1)).

This can be achieved, by implementing an additional criterion for the calculation, such as a classification according to the temporal dimension. Duckham et al. (2010) already considered this in their landmark weighting computation (Section 2.1.4). However, their weighting system was intended for the computation of the outdoor landmarks, but it can be still applicable to indoor environments. In terms of the spatial existence, they proposed *permanence* as a structural salience. They claim that objects, which are expected to change or move less frequently, are a better landmark candidate. As an example, they have compared toilets and service stations with schools and hospitals. In terms of the temporal existence, they introduced *night-time vs. daytime salience* as part of the visual salience, where it is defined that objects which are highly visible day and night are better landmark candidates. This factor could be of course adapted to the needs of the indoor environment. So it wouldn't be limited just to the day or night comparison, but pure temporal salience, which could be used in any time period.

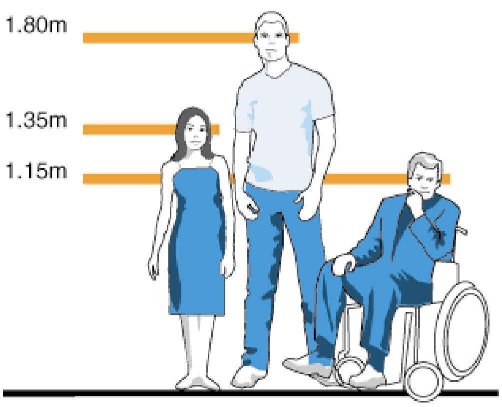
4.5 Agent's point of view

In 2D visibility-based navigation the environment is divided into regions – areas (Section 2.4). However in 3D space, we are dealing with the volume shaped regions (Chapter 3). Regions describe which objects are possible to perceive if one is located in them.

Since our aim is to navigate people, it needs to be realized that for a successful navigation we do not need all the regions, but only the ones that are located at the height of the user's eyes, because the user will actually be located only in these regions. In other words, a region located at 10 cm from the ground can be neglected, because no one looks at the environment from that perspective. The only exception is that the user's point of view varies with the height of the user, so it could happen that instructions which are adequate for a 1.80 m tall man are unsuitable for a person in a wheelchair (Figure 35(b)).



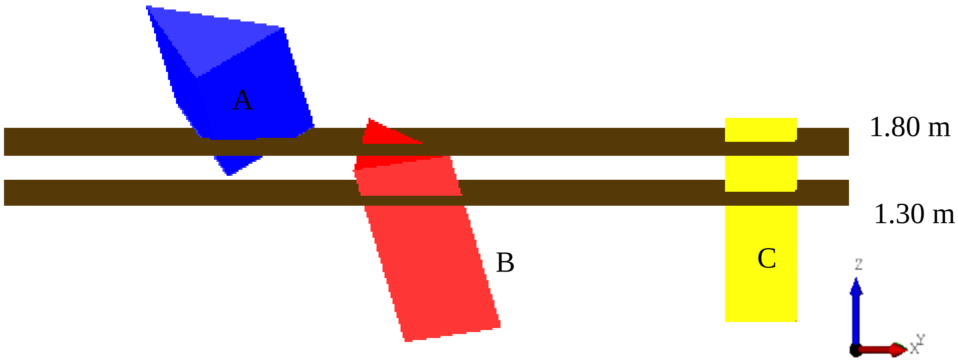
(a) Line of sight from two different heights.



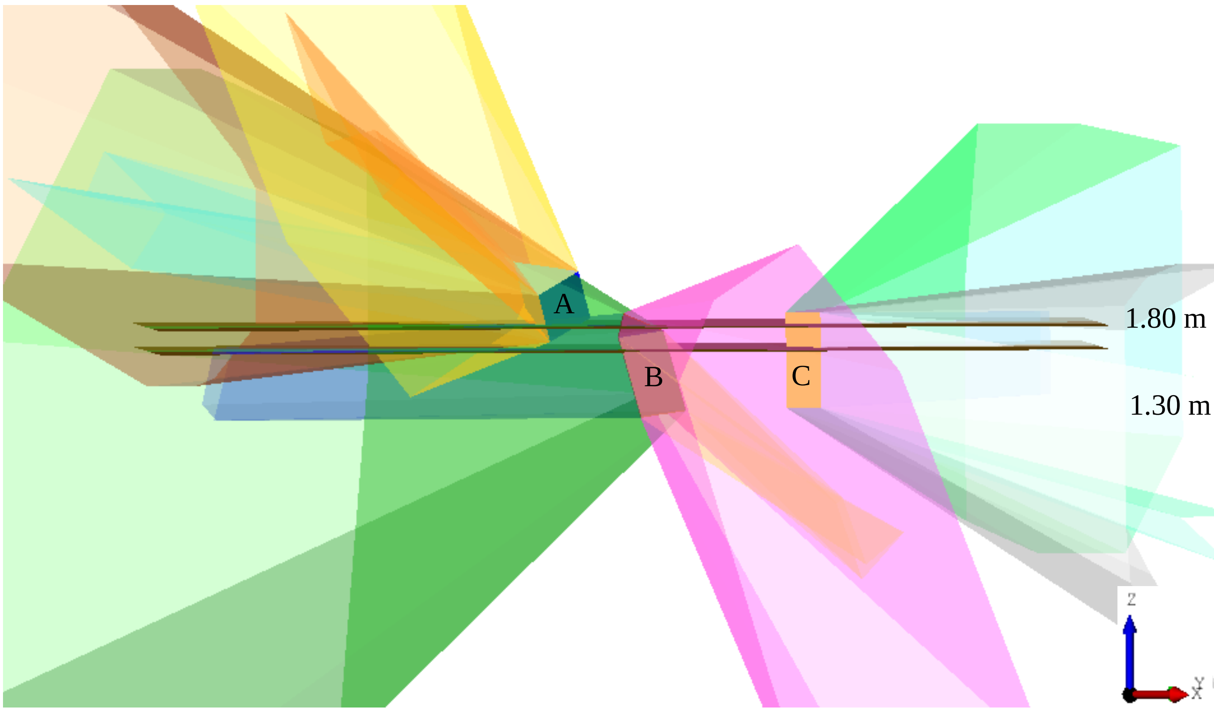
(b) Eye level of different people.

Figure 35: Different agent's point of view (Observer's_point_of_view 2016).

We do not have to compute the visibility in the whole 3D space, but only a slice of it, at a certain height, like presented in Figure 36(a). Therefore a map of external and internal volumes could be simplified, as shown in Figure 36(b).



(a) Possible eye level of the user, denoted with planes at 1.30 m and 1.80 m height.



(b) External and internal volumes computed from 3 objects (a), with the respective eye level at 1.30 m and 1.80 m.

Figure 36: We are only interested to extract volumes that are located at eye level.

Due to the fact that human height varies, we need to consider a broader slice than presented in Figure 36, from a person in a wheelchair to the tallest living man, whose height is around 2.50 m[4]. Despite the fact that the planes substantially reduce the volume of the regions, the difference between the two is still more than 1 m. This means that the panoramic view at the height of 1.20 m can still be different from the panoramic view at the height of 2.00 m, but it could be the same in certain cases.
To avoid this issue, such that the objects which are not visible along the whole height of the slice should be assigned a lower value, when calculating the adequacy of the objects as landmarks (Equation (1)) or they should even not be selected as a landmark. With this condition we would enable that the user will see them, independently from his height.

However, if we do not want to create an additional criteria when calculating suitability of landmarks in Equation (1), there are two options of how we could proceed and navigate the user:

1. Option: The user would provide his height[5] information. Furthermore, topological map would be extracted on that height and provide the user with the instructions, which are suited just for him.
2. Option: Altitude sections would be pre-calculated, which would highlight from which to which height we could navigate people based on the same instructions. For instance, the outcome of this step would be sectioned as 1.10 m – 1.40 m, 1.41 m – 2.00 m, 2.01 m – 2.50 m. They would denote three different groups of topological maps. In this option user would only choose one of the sections, which suits his height.

Option 1 might demand more on-the-fly load, whereas option 2 provides more general approach since it divides users into groups. In option 1 a planar topological map would be produced for individual person on a certain height. The produced topological map is the plane on which the user is walking and it resembles the topological map in 2D space in Figure 13(b). The only difference is that the regions have different labels since they represent visibility relations in 3D space. The region in this case illustrates the surface that user could be located and not the volume.

In order to provide sections described in option 2, it is required to produce a planar topological map for every 5 cm from 1.1 m height to 2.5 m height. With this step, we obtain 28 topological maps that has to be examined in order to see if they differ from each other. So each map should be compared with its successor. If the regions between them are changed, then the topological maps must be separated into two sections, otherwise they are combined. The process is presented in Figure 37.

[4] http://www.guinnessworldrecords.com/world-records/tallest-man-living

[5] It should be noted that the height of the agent does not match the height of the eyes. Therefore, a certain value still needs to be subtracted from the height of the agent to obtain the eye level.

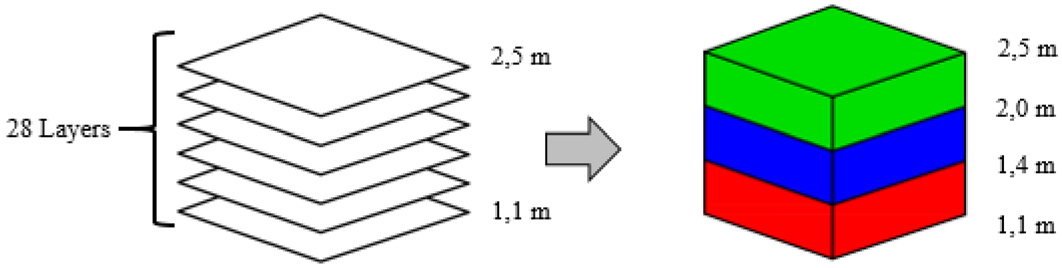


Figure 37: Option 2 presented in visual form. With 5 cm step (from 1.1 m to 2.5 m) 28 maps in 2D space are obtained. When comparing layers with its successor the change is detected in the height 1.40 m and 1.45 m. For that reason, all maps bellow and including 1.40 m are merged to one section (red colour). The process of comparing is continued until the last map is reached. In the end of this example 3 sections are obtained (red, blue green).

Option 2 could be used in case when there are not too many sections produced or it is not too complex to create them, otherwise it is better to use option 1.

4.6 Simplification of the regions

Until now, we have discussed many options that enable to simplify the visibility-based navigation. We preselected objects and simplified roles of some objects and took into account regions that matter only to the users with a fixed height. All of them contribute to a decreased number of regions. However, the space for improvement still exists, especially in the simplification of regions that represent subdivision of space.

Therefore, when we use the topological plane map (as described in Section 4.5), we can ignore or even erase some of the regions. In this case we are thinking of the regions that, because of their small size, are very hard to be met and can therefore be neglected. This way we could previously define the area threshold that the regions need to exceed in order to appear on the map.

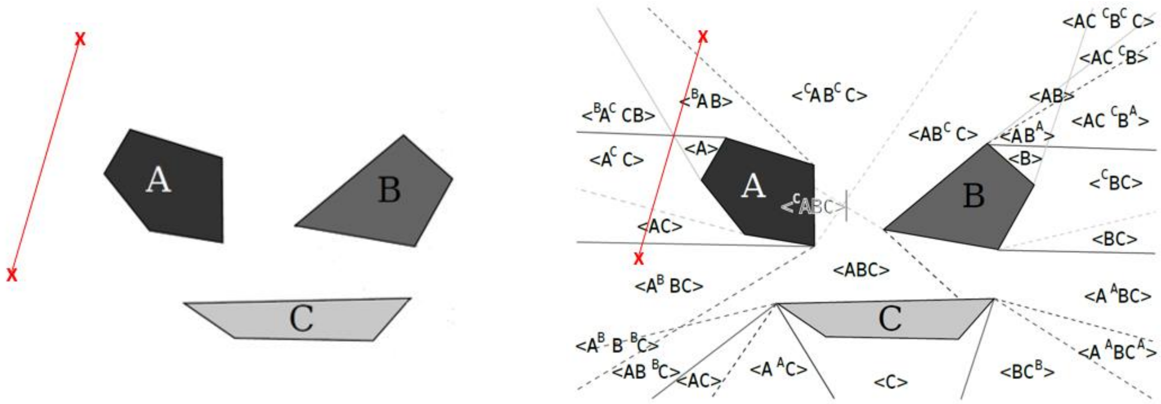
We could also determine the volume threshold before the production of the plane topological map. It would have a similar function as the plane topological map, determining the limit of which volumes can still be considered after the calculations and which cannot.

4.7 Merging of topologic and metric map

The shortest navigation route leading from the starting point to the end point could be until now computed in two ways, based on:

1. metric shortest path
2. topological shortest path on the graph

For simplicity, we will compare these two paths in a 2D example. **Metric shortest path** is a path which first computes the shortest path from A to B, like in Figure 38(a). Then the path is intersected with the visibility space of landmarks (Figure 38(b)) and then we generate natural language instructions.

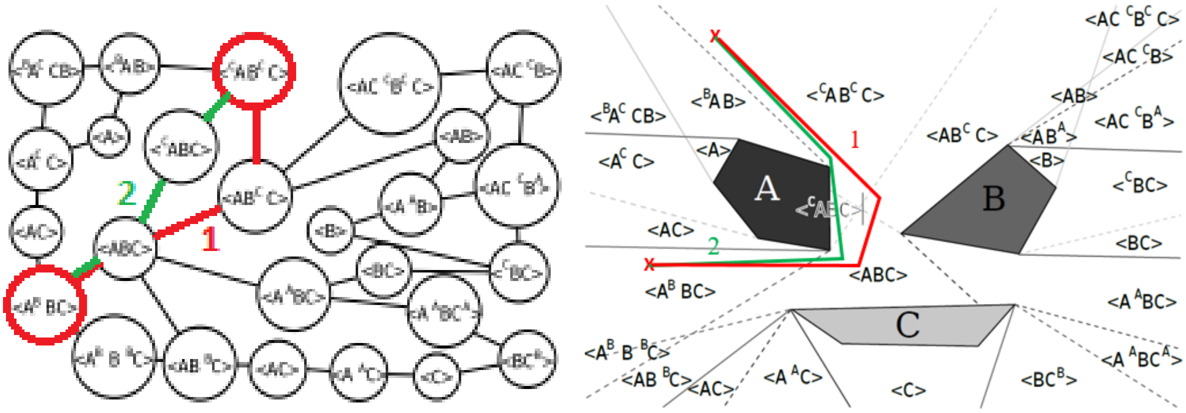


(a) Shortest metric path.

(b) Shortest metric path overlaped with visibility regions.

Figure 38: Shortest metric path computation (Fogliaroni et al. 2009).

Topological shortest path was already presented in Section 2.4 by Fogliaroni et al. (2009). It is computed as the shortest path on the graph. So if the user is located in region <A^B B C> and his destination is located in region <^{C}A B^C C>, in this case the shortest path on the graph has two solutions. One goes through <A B C> and <A B^C C>, like shown in Figure 39(a) with red line and the other through <A B C> and <^{C}A B C>, presented in Figure 39(a) with green line. When comparing Figure 39(b) and Figure 38(b) it is visible that the shortest metric path and the shortest topological path do not overlap, even more, they are passing through completely different routes. That is because the topological map presents the path that crosses the least amount of regions and its aim is to contain a smaller number of visibility-based instructions presented in Example 1. Arguably, such a path should be cognitively less demanding to be followed. Topological and metric maps also have similar role in 3D environment.



(a) Shortest topological path in the graph. (b) Shortest topological path presented in a metric map.

Figure 39: Shortest topological path computation (Fogliaroni et al. 2009).

At the moment topological and metric maps are seen as two separate things. However, it would be possible to merge them and have one map, which would contain information from both.

We suggest enriching the topological map with weights on its edges that denote the distance between the nodes. In the case of a 2D map every region should first have to have a determined centroid. Then one has to interconnect them like they are connected in the topological map, and measure the distance between them. The measured distance represents the value and is written into the edge of a weighted graph. At the end we obtain a graph as in Figure 40.

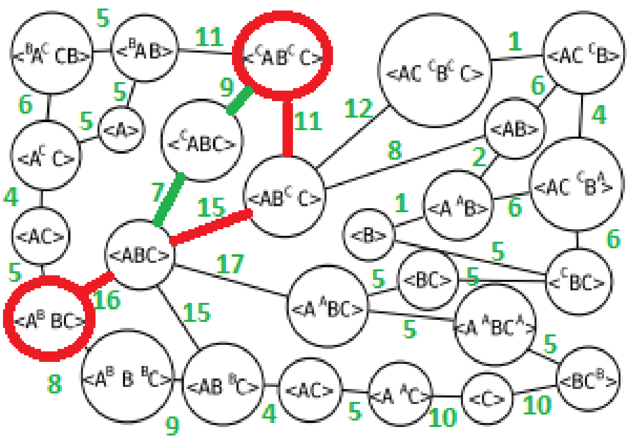


Figure 40: Topological map as weighed graph (Fogliaroni et al. 2009).

In this manner we could merge the data and thus offer a broader navigation to the user with the use of multiple criteria. So if we wanted to make the shortest metrical path, we would use a weighted topological map. And if we wanted to go on a path that is less complicated we would use the unweighted graph. Weights could also be used, when unweighted graph calculates more possible paths with the same amount of regions that need to be traversed like in Figure 40, where as a solution two lines are added, green and red. In this case we could take weights into account and thus select the shortest path. In this case, the green line would be the optimal one.

5 CONCLUSIONS, OPEN ISSUES AND FUTURE WORK

This work explores and analyses a solution to navigate people in indoor environments based on visibility relations in 3D space. This qualitative way of navigation based on visibility can be seen as an alternative to currently used navigation in the indoor environment.

Fogliaroni et al. (2009) has shown it is possible to localize and navigate agents in 2D environment based on the visibility information. We wanted to continue on this idea and analysed the extension to 3D space, where we could navigate people.

In this thesis we have described the process of navigation in 3D, where we took as a foundation 2D visibility navigation from Fogliaroni et al. (2009). As is 2D work of Fogliaroni et al. we have divided the environment into subregions to obtain quantitative coordinates. Instead of dealing with the external and internal tangent lines, we used external and internal tangent planes that provided us with external and internal volumes, which outline the sub regions of the environment. In comparison to 2D approach, in 3D we have obtained volumes instead of areas and faces instead of lines.

It is obvious that the transition from 2D space to 3D space makes everything more complex. Because of that we have to deal with new problems that do not occur in 2D. Therefore, we have defined the issues that need to be resolved before the implementation of 3D indoor landmark-based route instructions from the visibility model.

We have seen that the numbers of computed faces which describe the visibility regions are steeply rising with the number of objects which are included in the computation. Therefore, we proposed the usage of objects which are only necessary for the production of landmark-based route instructions, and so we will consequently reduce the load of computation and reduce the amount of regions.

One does not need to include all objects from the environment in the computation of the visibility. Only landmarks and obstacles need to be included, where landmarks are defined as the objects that we refer to in the route instructions and obstacles the objects which occlude the landmarks. Based on that definition we have proposed a filter that should be applied to the set of all objects in the environment - O. Filter is selecting the set of landmarks and set of obstacles from the set O. The determination of the obstacles depends largely on the size of the object as it is perceived from a viewpoint of the agent and the relative size of this projected obstacle with the respect to the size of the projected landmark. Since that is not obtained easily, we have suggested three criteria (position of the object, dimensions of objects in x, y, z directions, certain object groups) which could be applied to reduce the number of objects in the set of obstacles.

As a consequence of filtering we obtain only regions which are necessary for production of landmark-based route instructions from visibility model.

We proposed a special treatment of staircases and elevators in the computation of visibility, because they guide the user in the desired direction, therefore there is no need to look for landmarks or obstacles. Elevator even symbolizes a shortcut within the topological map, because it allows a connection of two nodes in the graph, which are otherwise not connected. To reduce the computational load and avoid unnecessary computations, we suggested exploiting of another measure which would take the advantage of the characteristics of the indoor space. We do not need to calculate the visibility between the objects which are not located in the same floor or room.

When calculating the salience of landmarks, the important thing is to include temporal dimension as a criterion, and so avoid the landmarks which are not fixed in time or place.

It is crucial to consider the agent's height when navigating based on visibility. We can not only develop one type of instructions, since a person in a wheelchair might see the landmarks differently than a very tall person. Therefore the agent's point of view needs to be taken into account. We propose that a map is made similar to 2D topological map while the labels would correspond to the 3D visibility relations.

Additional measures to simplify a topological map can be applied by ignoring or deleting some of the regions, with an area or volume smaller than some thresholds.

We have discussed the two shortest possible paths – metric and topological, which can be calculated from either a metric map or from a topological graph. We proposed to merge both of them into a weighted graph.

5.1 Future work

We realize that many challenges and opened questions still remain and due to their complexity, these will have to be put aside for future work.

5.1.1 Cutting function

In Chapter 3 the procedure to generate internal and external tangent planes was presented, which is performed by cutting function. This process is also performed in 2D space (Section 2.4), its role is to verify (i) if the tangent line is intersecting any objects between the points (Figure 11(a)), which are defining the tangent and (ii) if the tangent line doesn't cut through any objects. This quite trivial process becomes complex in 3D space because we do not operate with lines anymore, but with planes i.e. set of lines. The steps of cutting function in 3D were described already in Chapter 3, where we have written:

1. Check if plane intersects any other object in volume Between(B, C):
    a. If no: continue to second point
    b. If yes: It could be that this plane should not exist, or only a part of it should exist.
2. Obtain two half-planes, which are located in volumes Before(B, C) and After(B, C)
3. Limit each half-plane by the boundaries of other objects.

In Chapter 3 two steps are not completely defined, this are step 1.b and step 3.

Let us discuss step 1.b with an example presented in Figure 41, where we have two objects B and C (red and green cubes). Between them, we have an external tangent plane (orange), which is traversing through three vertexes v1, v2 and v3. If there is no object located between B and C, then we simply continue to the second point in our process. However, imagine there is a third object (blue cube) located between them (in volume Between(B, C)). Then we would not delete the tangential plane, like we would do in 2D with the tangent line. Instead we would trace only a part of a plane i.e. a plane would have a hole. The hole is denoted by the black colour in Figure 41.

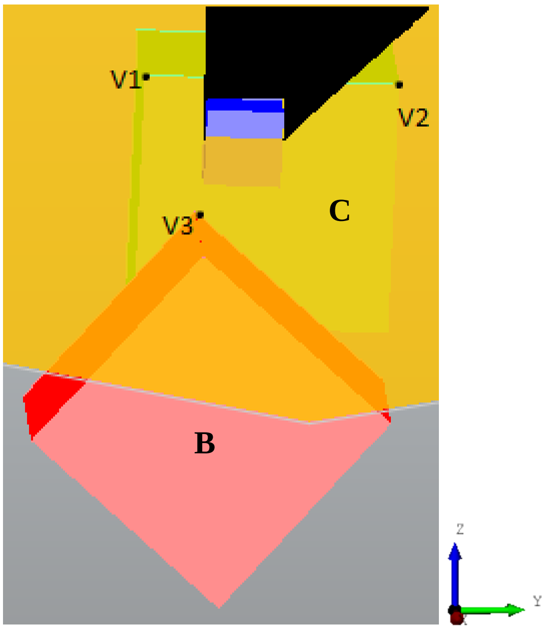


Figure 41: Cutting function in 3D for step 1b.

In the following we will discuss step 3, assuming we have already executed first two step successfully. Imagine we have two objects B and C (red and green cubes), between which we construct the external tangent plane (orange), denoted by vertexes v1, v2 and v3. Assume the third object (blue cube) is located behind the object C like in Figure 42. In that case a tangent plane does not stop at the third object, instead a third object creates a hole in it, which is in Figure 42 presented by the black colour.

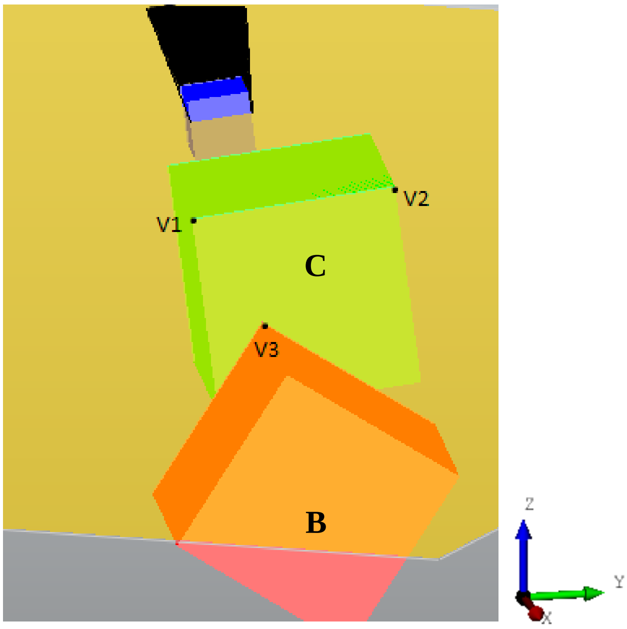


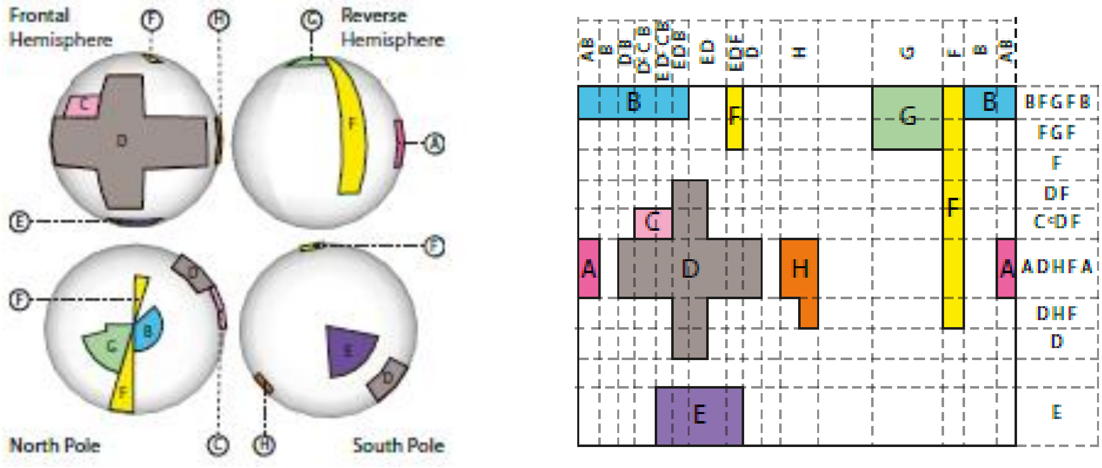
Figure 42: Cutting function in 3D for step 3.

In order to get a complete solution and to make sure that the theory corresponds to the praxis, this perspective still needs to be discussed and investigated further.

5.1.2 Panoramic view of the agent in 3D

Regions in visual based navigation are defined as zones where one can only observe certain objects. Therefore, they can be uniquely identified by recognizing the visibility relations. In 2D space, VCO was used for labelling each zone with a string representing visibility relations. This method cannot be directly applied in 3D since we have different visibility relations and therefore also combinations between them.

This topic was already investigated by Tassoni et al. (2011), where they proposed the use of a Visual Spherical Ordering (VSO) instead of VCO to identify a zone´s label. That means if the sphere is located in a zone, we could project all objects in a panoramic view on its surface. The obtained patterns represent the visibility relations for that zone. Moreover, it is possible to produce a VSO for every zone, based on the projection of the visible objects, like presented in Figure 43(a). In order to produce the labels from the sphere, they suggest projecting it to a planar surface by applying cylindrical projection. Furthermore, strings with which we identify every region are obtained by scanning the projection. To obtain a label, the projection is first scanned horizontally from bottom to top and then vertically from left to right. The string at the end is written by horizontal and vertical labels, separated with a semicolon. The 2D representation of the VSO is presented in Figure 43(b). The label of the example is as follows <AB, B, DB, DCCB, EDCCB, EDB, ED, EDF, D, H, G, F, B, AB; BFGFB, FGF, F, DF, C CDF, ADHFA, DHF, D, E>.



(a) Visual Spherical Ordering.

(b) Cylindrical projection and VSO labels.

Figure 43: 3D visibility and VSO (Tassoni et al. 2011).

This issue is still not completely resolved since their work only supports labels for a simple visibility model (Figure 19(a)), because the article was published before the finer 3D visibility relations (Section 2.5) were introduced. Therefore, the labels of Tassoni et al. need to be adapted to new relations (PartiallyVisibleLeft, PartiallyVisibleRight, PartiallyVisibleAboveLeft, PartiallyVisibleAboveRight, PartiallyVisibleBelowLeft, PartiallyVisibleBelowRight, PartiallyVisibleJoint). At the moment the label supports only PartiallyVisibleLeft (resp. PartiallyVisibleRight). If this relation holds, then the partially visible object is written as a superscript on the left (resp. right) side. In a case of a PartlyVisibleJoint, a superscript is written on both sides. To implement new relations in the labels, one could use superscripts for PartiallyVisibleAboveLeft, PartiallyVisibleAboveRight, subscripts for PartiallyVisibleBelowLeft, PartiallyVisibleBelowRight and middle scripts for PartiallyVisibleLeft, PartiallyVisibleRight (Figure 44). If, however the middle scripts turn out not to be distinctive enough, we can still add a dash alongside the middle script.

$$x^{a} \quad x_{a} \quad x\mathrm{a}$$

Figure 44: Superscript, subscript, middle script (Scripts 2015).

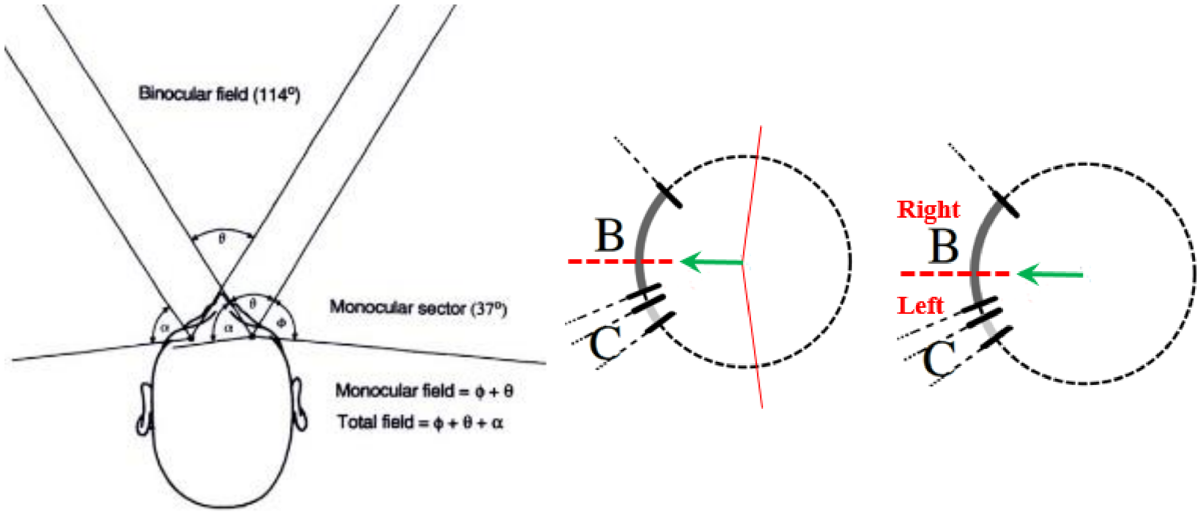
Moreover, their paper did not explain how to orient the sphere. Therefore, we suggest that the orientation of a sphere should match the orientation of gravity or as an alternative a vector pointing from feet to head, i.e. the same way as vector d is defined in a 3D visibility model – as it was discussed in Section 2.5.1.

Fogliaroni et al. (2009) have also considered a finer subdivision, in their work. The creation of a Light Zone in 2D space with the help of 5-intersection model, i.e. directional model, as it was discussed in Section 2.4 and presented in Figure 14. A similar step was not yet discussed in a 3D environment. But it would certainly be very useful, since at the moment, we only support relations, which declare that the landmark is visible. We do not know however, from which side the landmark is visible, therefore a directional model could be implemented in a 3D space as well. The only thing we have to consider here is how precise the breakdown of the directional model should be. Is it sufficient if we only define left-side, right-side and the between zones, or should we go into more detail and split them in planes Above, Bottom and Central as well? And after we've answered that question we still need to make sure that the regions are distinguished enough between each other. All this questions will need to be answered in the future.

5.1.3 Simplification of the instructions

As it was already discussed in Section 2.4, the instructions are produced based on the boundaries (lines in 2D or faces in 3D) the user is crossing, because each line (resp. face) represents two navigational behaviours which can be explained in natural language instructions.

Therefore, the instructions might be "walk around A keeping it on your right, until you don't perceive B anymore", where B could be located at the users' back. While these instructions work good for robotic agents with a 360 field of view, they are not very well suited for humans, since the user would constantly need to look back. The problem is, that people have a limited visual field (Figure 45(a)), which is defined as "an angle subtended at a point midway between the two eyes by all those points in space visible to either one eye or both" (Howard and Rogers 1995). For people the total visual field is usually 190˚ when the eyes are fixed. If however, we move them the angle value expends and becomes around 290˚ (Howard and Rogers 1995).



(a) Total visual field of a human (Howard and Rogers 1995).

(b) 190˚ visual field in VCO (Fogliaroni et al. 2009).

(c) Separating VCO to left and right side, based on the orientation (Fogliaroni et al. 2009).

Figure 45: Total visual field of human.

To avoid these circumstances, we should consider the direction of the user’s movement and integrate in the instructions the landmarks that are in the user’s sight. For that purpose, VCO (resp. VSO) properties could be exploited, meaning that we should consider one’s orientation when navigating and find a solution to only mention landmarks in the instructions that are at their line of sight. Like presented in Figure 45(b), red line marks the visibility field based on the user’s orientation which is presented as green line. All the landmarks behind the line should be neglected when producing the route instructions. But the consequences of this simplification should be researched in the future. For example, if a person would travel from a region <ABC> to <A^BBC> (marked in Figure 46), the object B would be located at the users back. So if we would consider a visual field of 190˚ (Figure 45(b)) and ignore this object the instructions might not change. In the future it should be investigated if this change influences the course of visibility-based navigation and if it would even make sense to apply it.

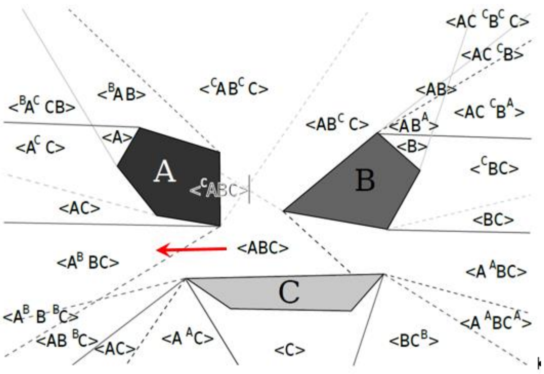


Figure 46: Navigation from a region <ABC> to <A^BBC> (Fogliaroni et al. 2009).

By applying user's orientation in VCO (resp. VSO), it would also be possible to cut the VCO (resp. VSO) on the right and the left side, as presented in Figure 45(c) which could provide extra information to the user.

5.1.4 Transparent objects

Until now we have always assumed that the objects were opaque and one cannot see through them. Therefore, they represent a physical obstacle – obstacle for a movement in space and a visibility obstacle – obstacle for the user's line of sight. But there are transparent objects, which are quite common in the indoor environment, like glass walls or doors. These objects only represent the physical obstacle since they do not occlude user's line of sight (Figure 47), which creates unusual consequences in the topological graph. As depicted in Figure 47, physically we have two regions A and B, i.e. either the user is located at one side of the wall or at the other. However, user's view is the same on both sides. For now, everyone neglected this issue, however it will need resolving in the future.

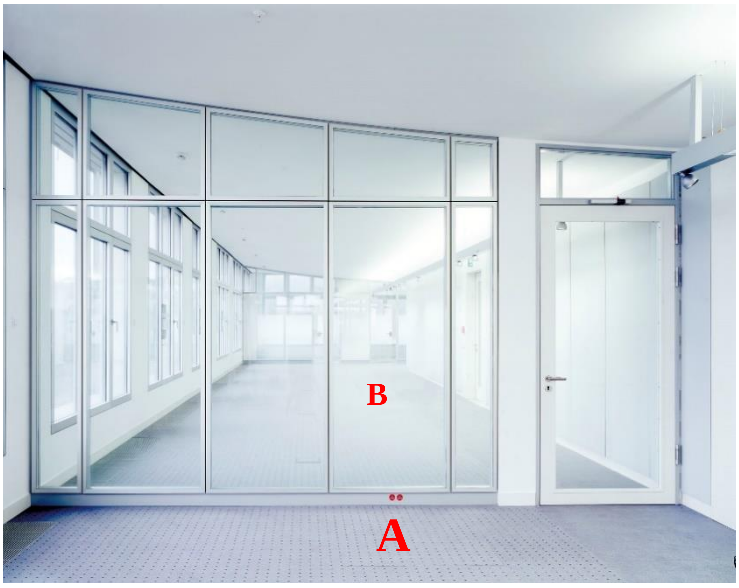


Figure 47: Transparent obstacle (Glass_door 2016).

6 REFERENCES

Allen, Gary L. 1999. “Spatial Abilities, Cognitive Maps, and Wayfinding.” *Wayfinding Behavior: Cognitive Mapping and Other Spatial Processes* 46–80.

Amemiya, Tomohiro, Jun Yamashita, Koichi Hirota, and Michitaka Hirose. 2004. “Virtual Leading Blocks for the Deaf-Blind: A Real-Time Way-Finder by Verbal-Nonverbal Hybrid Interface and High-Density RFID Tag Space.” *Proceedings - Virtual Reality Annual International Symposium* 165–72.

Anon. 2016. “Multipath Propagation.” 75. Retrieved October 19, 2016 (http://www.iitg.ernet.in/scifac/qip/public_html/cd_cell/chapters/a_mitra_mobile_communication/chapter5.pdf).

Arikawa, Masatoshi, ShiN’ichi Konomi, and Keisuke Ohnishi. 2007. “NAVITIME: Supporting Pedestrian Navigation in the Real World.” *IEEE Pervasive Computing* 6(3):21–29.

Beal, J. R. 2003. “Contextual Geolocation, a Specialized Application for Improving Indoor Location Awareness in Wireless Local Area Networks.” *MICS2003: The 36th Annual Midwest Instruction and Computing Symposium, Duluth, Minnesota*.

Billen, R. and E. Clementini. 2004. “A Model for Ternary Projective Relations between Regions.” *Advances in Database Technology-EDBT 2004* 537–38.

Burnett, Gary E. and Kate Lee. 2005. “The Effect of Vehicle Navigation Systems on the Formation of Cognitive Maps.” Pp. 407–18 in *International Conference of Traffic and Transport Psychology*.

Caduff, David and Sabine Timpf. 2008. “On the Assessment of Landmark Salience for Human Navigation.” *Cognitive Processing* 9(4):249–67.

Clementini, Eliseo and Roland Billen. 2006. “Modeling and Computing Ternary Projective Relations between Regions.” *IEEE Transactions on Knowledge and Data Engineering* 18(6):799–814.

Cross_Section. 2016. “Cross Section.” Retrieved November 28, 2016 (https://static.dezeen.com/uploads/2015/10/Andarzgoo-residential-building_Ayeneh-Office_dezeen_1.gif).

Dijkstra, Edsger W. 1959. “A Note on Two Problems in Connexion with Graphs.” *Numerische Mathematik* 1(1):269–71.

Duckham, Matt, Stephan Winter, and Michelle Robinson. 2010. “Including Landmarks in Routing Instructions.” *Journal of Location Based Services* 4(1):28–52.

Fallah, Navid, Ilias Apostolopoulos, Kostas Bekris, and Eelke Folmer. 2013. “Indoor Human Navigation Systems : A Survey.” *Interacting with Computers* 25(1):4–8.

Fang, Zhixiang, Qingquan Li, Xing Zhang, and Shih-Lung Shaw. 2012. “A GIS Data Model for Landmark-Based Pedestrian Navigation.” *International Journal of Geographical Information Science* 26(5):817–38.

Floor_plan. 2016. “Floor Plan.” Retrieved November 25, 2016 (http://www.selldorf.com/drawings/brown-university-john-hay-library).

Fogliaroni, Paolo and Eliseo Clementini. 2014. “Modeling Visibility in 3D Space : A Qualitative Frame of Reference.” *9th International 3DGeoInfo 2014 - Lecture Note in Geoinformation and Cartography* 1–19.

Fogliaroni, Paolo, S. Shekhar, H. Xiong, and X. Zhou. 2016. “Visibility Relations.” *Encyclopedia of Geographic Information Science (2nd Edition).* 8.

Fogliaroni, Paolo, Jan Oliver Wallgrün, Eliseo Clementini, Francesco Tarquini, and Diedrich Wolter. 2009. “A Qualitative Approach to Localization and Navigation Based on Visibility Information.” *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* 5756 LNCS:312–29.

FreeCAD. n.d. “PythonOCC - FreeCAD Documentation.” Retrieved October 7, 2016 (http://www.freecadweb.org/wiki/index.php?title=PythonOCC).

Galton, Antony P. 1994. “Lines of Sight.” *AI and Cognitive Science ’94* (March):103–13.

Gerstweiler, Georg, Emanuel Vonach, and Hannes Kaufmann. 2015. “HyMoTrack: A Mobile AR Navigation System for Complex Indoor Environments.” *Sensors (Switzerland)* 16(1):1–19.

Glass_door. 2016. “Glass Door.” Retrieved October 5, 2016 (http://gibo2006.en.made-in-china.com/productimage/oMXQqjYdQCpN-2f1j00tMpEaHdrsmul/China-Fixed-Glass-Partition-Wall.html).

Golledge, RG. 1999. “Human Wayfinding and Cognitive Maps.” *Wayfinding Behavior: Cognitive Mapping and Other Spatial Processes* 5–45.

Harle, Robert. 2013. “IEEE Xplore - A Survey of Indoor Inertial Positioning Systems for Pedestrians.” *IEEE Communications Surveys \& Tutorials* 15(3):1281–93.

Hart, Peter E., Nils J. Nilsson, and Bertram Raphael. 1968. “A Formal Basis for the Heuristic Determination of Minimum Cost Paths.” *IEEE Transactions on Systems Science and Cybernetics* 4(2):100–107.

Hightower, J. and G. Borriello. 2001. “Location Sensing Techniques.” *IEEE Computer* 34(july):1–8.

Hohenschuh, Frank. 2004. “Prototyping Eines Mobilen Navigationssystems Für Die Stadt Hamburg.” Universität Hamburg.

Holiday_Season. 2016. “Holiday Season.” Retrieved October 7, 2016 (https://www.colourbox.com/image/a-large-shopping-mall-center-in-berlin-decorated-for-christmas-shopping-time-with-glistening-lights-and-trees-image-2702279).

Howard, Ian P. and Brian J. Rogers. 1995. *Binocular Vision and Stereopsis*.

Hund, Alycia M. and Jennifer L. Minarik. 2009. “Getting From Here to There: Spatial Anxiety, Wayfinding Strategies, Direction Type, and Wayfinding Efficiency.” *Spatial Cognition and Computation: An Interdisciplinary Journal* 6(3):181.

Hund, Alycia M. and Amanda J. Padgitt. 2010. “Direction Giving and Following in the Service of Wayfinding in a Complex Indoor Environment.” *Journal of Environmental Psychology* 30(4):554–55.

Kennedy, Lyndon S. and Mor Naaman. 2008. “Generating Diverse and Representative Image Search Results for Landmarks.” Pp. 297–306 in *Proceedings of the 17th international conference on World Wide Web*.

Klippel, Alexander and Stephan Winter. 2005. “Structural Salience of Landmarks for Route Directions.” Pp. 347–62 in *International Conference on Spatial Information Theory*.

Köhler, C. 2002. “The Occlusion Calculus.” *Cognitive Vision Workshop* 420–50.

Lamps1. 2016. “Lamps1.” Retrieved November 30, 2016 (http://www.applegazette.com/feature/inside-apple-hq/).

Lamps2. 2016. “Lamps2.” Retrieved November 30, 2016 (https://www.pinterest.com/pin/467249148562112l2/).

Lawton, Carol A. and Janos Kallai. 2002. “Gender Differences in Wayfinding Strategies and Anxiety about Wayfinding: A Cross-Cultural Comparison.” *Sex Roles* 47:389–401.

Levitt, Tod S. and Daryl T. Lawton. 1990. “Qualitative Navigation for Mobile Robots.” *Artificial Intelligence* 44(3):305.

Lygin, Roman. 2009. “Open CASCADE Notes: Topology and Geometry in Open CASCADE. Part 1.” Retrieved October 7, 2016 (http://opencascade.blogspot.si/2009/02/topology-and-geometry-in-open-cascade.html).

Mautz, Rainer. 2012. “Indoor Positioning Technologies.” *Institute of Geodesy and Photogrammetry, Department of Civil, Environmental and Geomatic Engineering, ETH Zurich* (February 2012):127.

Michon, Pierre-Emmanuel and Michel Denis. 2001. “When and Why Are Visual Landmarks Used in Giving Directions?” *Spatial Information Theory* 2205:292–305.

Montello, Daniel R. 1998. “A New Framework for Understanding the Acquisition of Spatial Knowledge in Large-Scale Environments." *Spatial and Temporal Reasoning in Geographic Information Systems* 143.

Montello, Daniel R. 2009. "Cognitive Research in GIScience: Recent Achievements and Future Prospects." *Geography Compass* 3(5):1824–40.

Montello, Daniel R. and Martin Raubal. 2013. "Functions and Applications of Spatial Cognition." *The APA Handbook of Spatial Cognition* 555–91.

Montello, DR. 2001. "Spatial Cognition." *International Encyclopedia of the Social and Behavioral Sciences*.

Montello, DR. 2005. "Navigation." Pp. 257–94 in *Handbook of Visuospatial Thinking*. Cambridge University Press.

Mulloni, Alessandro, Daniel Wagner, Istvan Barakonyi, and Dieter Schmalstieg. 2009. "Indoor Positioning and Navigation with Camera Phones." *IEEE Pervasive Computing* 8(2):22–31.

Observer's_point_of_view. 2016. "Observer's Point of View." Retrieved October 1, 2016 (http://universaldesign.ie/Technology-ICT/Irish-National-IT-Accessibility-Guidelines/Public-Access-Terminals/Public-access-terminals-Priority-1/1-2-Ensure-that-displays-are-within-sight-of-people-of-all-heights-and-people-sitting-in-a-wheelchair-or-buggy/).

Ohm, Christina, Manuel Müller, and Bernd Ludwig. 2015. "Displaying Landmarks and the User ' S Surroundings in Indoor Pedestrian Navigation Systems." *Journal of Ambient Intelligence and Smart Environments* 7(5):635–57.

Ohm, Christina, Manuel Müller, Bernd Ludwig, and Stefan Bienk. 2014. "Where Is the Landmark? Eye Tracking Studies in Large-Scale Indoor Environments." *CEUR Workshop Proceedings* 1241:47–51.

OpenCascade. 2016a. "Open CASCADE Technology: Modeling Data." Retrieved October 7, 2016 (https://www.opencascade.com/doc/occt-6.7.0/overview/html/user_guides__modeling_data.html).

OpenCascade. 2016b. "Open CASCADE Technology: Overview." Retrieved October 7, 2016 (https://dev.opencascade.org/doc/overview/html/index.html).

OpenCascade. n.d. "Open CASCADE Technology: BRep Format." Retrieved October 7, 2016 (https://www.opencascade.com/doc/occt-6.7.0/overview/html/occt_brep_format.html#occt_brep_format_6).

Ornament. 2016. "Ornament." Retrieved October 7, 2016 (https://beingafreelanceartist.com/tag/origami-2/).

Poster. 2016. "Poster." Retrieved October 1, 2016 (http://www.deltamedias.com/portfolio/displaystands/).

Radoczky, Verena. 2003. “Kartographische Unterstützungsmöglichkeiten Zur Routenbeschreibung von Fußgängernavigationssystemen Im In- Und Outdoorbereich.” TU-Wien.

Randell, David A., Zhan Cui, and Anthony G. Cohn. 1992. “A Spatial Logic Based on Regions and Connection.” *3rd International Conference On Knowledge Representation And Reasoning* 92:165–76.

Randell, David and Mark Witkowski. 2002. “Building Large Composition Tables Via Axiomatic Theories.” *{KR2002}: Principles of Knowledge Representation and Reasoning* 1:26–35.

Randell, David, Mark Witkowski, and Murray Shanahan. 2001. “From Images to Bodies: Modelling and Exploiting Spatial Occlusion and Motion Parallax.” *IJCAI International Joint Conference on Artificial Intelligence* 57–66.

Raubal, Martin and Stephan Winter. 2002. “Enriching Wayfinding Instructions with Local Landmarks.” Pp. 243–59 in *International Conference on Geographic Information Science*.

Retscher, Günther and Michael Thienelt. 2004. “NAVIO-- A Navigation and Guidance Service for Pedestrians.” *Journal of Global Positioning Systems* 3(1–2):208–17.

Richter, Kai-Florian. 2013. “Prospects and Challenges of Landmarks in Navigation Services.” Pp. 83–97 in *Cognitive and Linguistic Aspects of Geographic Space*.

Richter, Kai-Florian and Stephan Winter. 2011. “Harvesting User-Generated Content for Semantic Spatial Information: The Case of Landmarks in OpenStreetMap.” *Proceedings of the Surveying & Spatial Sciences Biennial Conference, Wellington, New Zealand* 2125:75–86.

Richter, Kai-Florian and Stephan Winter. 2014. “Landmarks: GIScience for Intelligent Services.” Retrieved September 2, 2016 (https://books.google.at/books?hl=en&lr=&id=i-i5BAAAQBAJ&oi=fnd&pg=PR7&ots=pMlL4bpgn0&sig=akTppyzSFknSAqFTeiT19Njoohw).

Richter, Kf, Drew Dara-Abrams, and M. Raubal. 2010. “Navigating and Learning with Location Based Services: A User-Centric Design.” *Proceedings of the 7th International Symposium on LBS and Telecartography* 261–76.

Sadalla, EK and SG Magel. 1980. “The Perception of Traversed Distance.” *Environment and Behavior* 12(1):65–79.

Sadalla, EK and LJ Staplin. 1980. “An Information Storage Model for Distance Cognition.” *Environment and Behavior* 12(2):183.

Scripts. 2015. “No Title.” Retrieved October 7, 2016 (http://tex.stackexchange.com/questions/250961/subscript-superscript-middlescript).

Sefelin, Reinhard et al. 2005. “Landmarks: Yes; but Which?: Five Methods to Select Optimal Landmarks for a Landmark-and Speech-Based Guiding System.” *Proceedings of the 7th International Conference on Human Computer Interaction with Mobile Devices & Services - MobileHCI ’05* (January 2005):287–90.

Shopping_Mall. 2016. “Shopping Mall.” Retrieved October 1, 2016 (http://els-bg.com/en/zavarsheni-proekti).

Siegel, AW and SH White. 1975. “The Development of Spatial Representations of Large-Scale Environments.” *Advances in Child Development and Behavior* 10:9–55.

Staircase. 2016. “Staircase.” Retrieved September 29, 2016 (http://www.coroflot.com/joeluckettdesign/Staircase-Working-Drawings).

Tarquini, F., G. De Felice, P. Fogliaroni, and E. Clementini. 2007. “A Qualitative Model for Visibility Relations.” *Lecture Notes in Computer Science* 4667:510–13.

Tassoni, Sergio, Paolo Fogliaroni, Mehul Bhatt, and Giorgio De Felice. 2011. “Toward a Qualitative 3D Visibility Model.” *25th International Workshop on Qualitative Reasoning, Co-Located with the IJCAI-11 Conference, Barcelona, Spain*.

Tezuka, Taro and Katsumi Tanaka. 2005. “Landmark Extraction: A Web Mining Approach.” Pp. 379–96 in *International Conference on Spatial Information Theory*.

Thorndyke, PW. 1981. “Distance Estimation from Cognitive Maps.” *Cognitive Psychology* 13(4):526–50.

Tom, Ariane and Michel Denis. 2003. “Referring to Landmark or Street Information in Route Directions: What Difference Does It Make?” *Spatial Information Theory. Foundations of Geographic Information Science* 362–74.

Viaene, Pepijn, Pieter Vansteenkiste, Matthieu Lenoir, Alain De Wulf, and Philippe De Maeyer. 2016. “Examining the Validity of the Total Dwell Time of Eye Fixations to Identify Landmarks in a Building.” *Journal Of Eye Movement Research* 9(3):1–11.

Vilar, Elisângela, Francisco Rebelo, and Paulo Noriega. 2014. “Indoor Human Wayfinding Performance Using Vertical and Horizontal Signage in Virtual Reality.” *Human Factors and Ergonomics in Manufacturing & Service Industries* 24(6):603.

Waller, David and Lynn Nadel. 2013. “Handbook of Spatial Cognition” edited by D. Waller and L. Nadel.

Wang, Ranxiao Frances and Elizabeth S. Spelke. 2000. “Updating Egocentric Representations in Human Navigation.” *Cognition* 77(3):215–50.

Weisman, J. 1981. “Evaluating Architectural Legibility: Way-Finding in the Built Environment.”

Environment and Behavior 13(2):189–204.

Wither, Jason, Carmen E. Au, and Radek Rischpater, Raymond Grzeszczuk. 2013. “Moving beyond the Map: Automated Landmark Based Pedestrian Guidance Using Street Level Panoramas.” Pp. 203–12 in *Proceedings of the 15th international conference on Human-computer interaction with mobile devices and services*.

Wolfensberger, Marius and Kai-Florian Richter. 2015. “A Mobile Application for a User-Generated Collection of Landmarks.” Pp. 3–9 in *International Symposium on Web and Wireless Geographical Information Systems*.

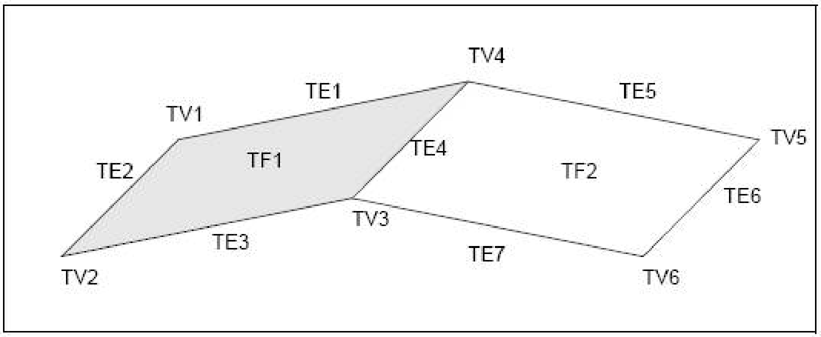
7 APPENDIX

7.1 Data Processing Tools

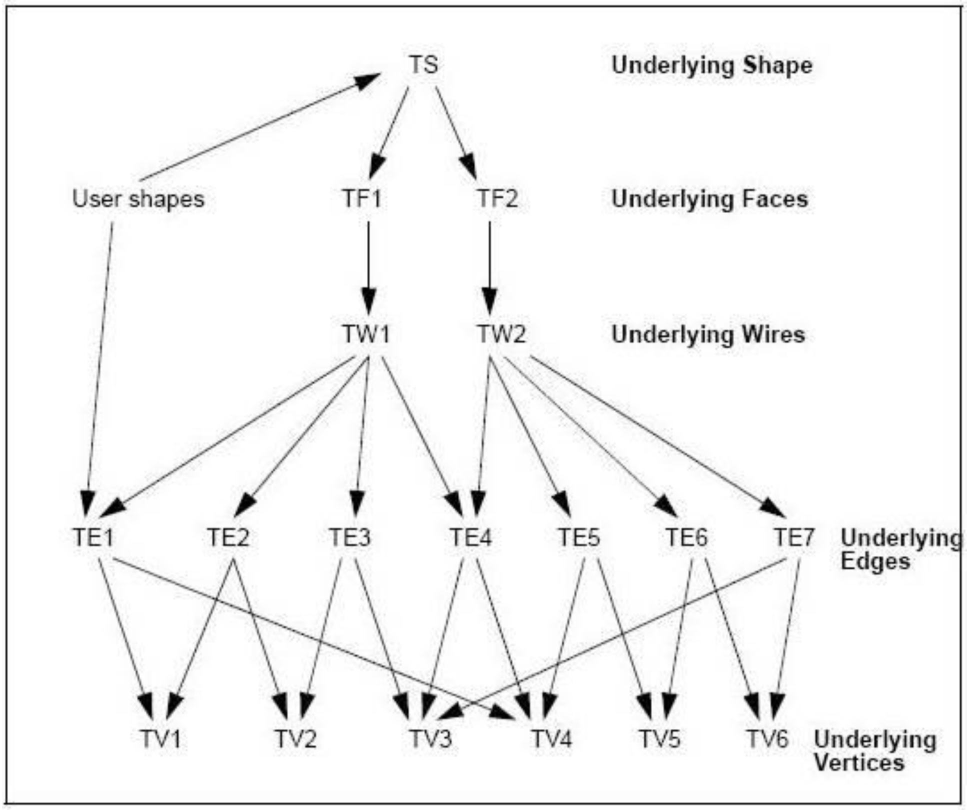
In this work, we have provided several figures (Figure 20, Figure 21, Figure 22, etc.) of 3D visibility model in order to visualize the computed visibility regions between objects in the environment. They were generated with work-in-progress algorithm, which will be published soon. For that purpose, we have used Open CASCADE Technology (OCCT), which is a software development platform providing services for 3D modelling and visualization. Most of OCCT functionality is written in C++ language (OpenCascade 2016b). For our purposes we have used PythonOCC, which aims to provide the whole range of Open Cascade (OCC) classes and functions into a python language (FreeCAD n.d.). We have used OCC also because it supports one main standard for 3D models – Industry Foundation Classes (IFC).

OCC supports Boundary Representation (BRep) format, which is used to store 3D models, that consists of vertices, edges, wires, faces, shells, solids, space location and orientation (OpenCascade n.d.).

OCC topology follows the STEP standard ISO-10303-42, where the structure is an oriented one-way graph, where parents refer to their children, and there are no references to the parents (Lygin 2009). Such graph is presented in Figure 48(b). It is visible there is a shape TS (Figure 48(a)), which consists of two faces (TF1, TF2). The latter are constructed from wires (TW1, TW2), where each wire consists of edges and edges consists of vertices. The topology structure is connected with geometry (BRep).



(a) Shape TS.



(a) Oriented one-way graph.

Figure 48: Data structure in OCC represented in the example (OpenCascade 2016)

Here are presented the basic types:

- **Vertex** corresponds to a point in 3D space. If we create a box, it will contain 48 vertices, where only 8 would contain a unique geometric coordinates.
- **Edge** is a topological entity that may designate a face boundary. In a box, there would be 24 edges, because each of 6 faces would be constructed from 4 edges.
- **Face** is a topology entity that describes a boundary unit of the 3D body. A face is described with its underlying surface and one or more wires. For instance, a solid cylinder consists of 3 faces – bottom and top, and lateral.