Student thesis series INES nr 491

Automated building compliance checking: emphasis on building placement and parking availability

Min Ya Zhang

2019
Department of
Physical Geography and Ecosystem Science
Lund University
Sölvegatan 12
S-223 62 Lund
Sweden

Min Ya Zhang
Automated Building Compliance Checking: Emphasis on Building Placement and Parking Availability
Master degree thesis, 30 credits in *Geomatics*
Department of Physical Geography and Ecosystem Science, Lund University

Level: Master of Science (MSc)

Course duration: *January* 2019 until *June* 2019

Disclaimer

This document describes work undertaken as part of a program of study at the University of Lund. All views and opinions expressed herein remain the sole responsibility of the author, and do not necessarily represent those of the institute.

Automated Building Compliance Checking: Emphasis on Building Placement and Parking Availability

Min Ya Zhang

Master thesis, 30 credits, in *Geomatics*

Per-Ola Olsson
Lund University

Exam committee:
Lars Harrie, Lund University
Olive Niyomubyeyi, Lund University

Abstract

Building permit is an important aspect of urban planning and an essential part of the construction industry. In recent years, building permit automation has become a rising international topic that has been studied by various institutes and organizations. Not only is automated compliance checking convenient and efficient, it is also proven to be cost effective and environmentally friendly. This study addresses four aspects of the building permit requirements: building height, roof angle, building placement and parking availability, with emphasis on the latter two. These requirements are regulated in the local municipality's detailed development plans. This study successfully automates the compliance checking of building placement and parking availability using various spatial and mathematical transformations, as well as minimum distance logic. The results from this study demonstrate that it is possible to automate the compliance checking for certain building requirements in Sweden.

Keywords: building permission, building requirements, compliance checking, house placement, parking availability

Acknowledgements

Throughout the writing of this thesis, I have received a lot of assistance from my supervisor Dr.Per-Ola Olsson from the Department of Physical Geography and Ecosystem Science at Lund University. Per-Ola used his expertise in GIS and city modelling to help me solve difficult situations and guide me in the right direction. He has provided me support and assistance whenever I needed. I would like to thank him for all the support he has provided me to help me achieve the goals in this thesis.

I would like to acknowledge Simon Jönsson, imaging engineer from Axis Communications. Simon tested my algorithm multiple times to find faults and weaknesses. Because of his effort, I was able to develop an algorithm that can handle various situations. I would like to thank him for his brilliant testing skills and constant support.

Table of Contents

1. Introduction

1.1 Background

Urban planning is a process that is essential for the development and maintenance of infrastructures and land use in a region. This process is broad and extends to both residential and recreational areas, directly affecting all buildings on these lands. To ensure that the construction, demolition and alteration of buildings are safe and legal, building permits became a necessity. Most building permit applications include the drawing of floor plans, site plan, elevations and cross sections to scale in 2D (Permits and Processes, 2019). These drawings are either submitted in paper or pdf format. To manually check if the building structures meet the requirements in the detailed development plan, scale ruler is used for paper version and program measurement tools are used for pdf version. However, the rule checking process can be time consuming since all properties of the building must be manually checked. Throughout the last decade, demands for automating the building permit process has become increasingly important to cope with fast urban developments.

An important part of building permit automation is digitization of the nearby built environment. It is not sufficient to only digitize the building, the surrounding areas must also be digitized. This is because regulations for constructions is relevant to the surrounding environment, for example, proximity to road or green area. A digitized neighborhood may contain different layers of information: buildings, roads, public green area, buildable area, non-buildable area, etc. These features are closely related to different aspects of building requirements, and the key to determining their relationship is through digitization of the nearby built environment. Digitized documents that are used for planning include 2D maps, 3D city models and detailed development plans. Digitization allows these data to be re-used numerous times until changes are made to the environment, this saves resources as the data are stored and monitored through time.

In general, building permit automation saves time and resources, which speeds up the process for building construction and allows local governments to allocate the resources to other social sectors. Successful optimization of the building checking process improves efficiency and benefits a large number of countries across the world. Countries that are ahead in building permit automation have shorter building permit application time, thus contributing to a more efficient construction process (Doing Business Training for Reform, 2019). This is essentially important for countries or cities that are undergoing immediate housing crisis.

In Sweden, the building permit process involves various laws and regulations on a national and regional level. The Planning and Building Act advocates for a balance between good living conditions and a sustainable environment with consideration of both public and private interest (Planning and Building Act (2010:900) 2010). The Planning and Building Act regulates planning on a national scale, while the detailed development plans function on a local scale by allowing municipalities to regulate the usage of land and water within its geographical boundary (Planning process, 2018). The detailed development plans are a big part of the building permit process because they also regulate different properties of a building, such as height, area and location (Planning process, 2018). While all aspects of building requirements are important, this study only addresses four of them: building height, roof angle, house placement and parking availability. This study mainly focuses on: 1) the placement of the building and its distance from the property boundaries. 2) availability of parking space in a property with regards to house placement. If these regulations are not met, then construction is not allowed even if the area is considered buildable. It is therefore crucial to check different aspects of building regulations to ensure that the building meets all requirements of the detailed development plan.

1.2 Aim

Researchers have focused on both general and specific parts of building permit automation; this study aims to contribute by focusing on two specific parts of the rule checking process. The overall aim of this study is to develop a method to automate the checking of building placement and parking availability to ensure compliance to the municipality’s detailed development plans, as an effort to optimize parts of the building permit checking process. This study focuses on developing efficient ways to automate the following regulations according to the detailed development plan:

- Is the house location according to regulations?
- Is the minimum distance to property boundaries within legal distance?
- Is there available parking space with accordance to house location?

1.3 Disposition

This study is divided into 7 Sections: Section 1 describes the underlying problems, main purpose, and aim of the study. Section 2 provides background information regarding the current building permit situation both internationally and nationally. These background information include literature reviews from related studies as well as proposed reformations. Section 3 outlines the methodologies used to develop the algorithm in this study. The methodology section focuses on building placement and parking availability automation. Section 4 describes the tools that are used as well as the algorithm workflow. The workflow is divided into two parts, representing two processes: building placement and parking availability. Section 5 includes the case study for Höganäs municipality. For the case study, the algorithm performs 20 tests to validate the algorithm's accuracy, the results of those tests are also in section 5. Section 6 discusses potential improvements and future functionality of the algorithm. Section 7 provides an overview of the study and concludes the outcomes of the results.

2. Literature Review

2.1 Current Status of the Building Permit Process

2.1.1 Singapore

Automation of building permit compliance checking is an international topic that has been studied internationally. One of the world's leading countries in construction is Singapore. Singapore's Building and Construction Authority (BCA) agency aims to build safe, high quality and sustainable structures and infrastructures. BCA launched an automated plan checking system called Construction Real Estate Network (CORENET) in 1995. The sole purpose of the CORENET system is to increase the efficiency of the building permission process. In 2000, the system introduced electronic submissions of building plans and provided information regarding building regulations. The CORENET system significantly benefits the construction industry by allowing building professionals to submit building plans from anywhere in the world at anytime. Since the launch, the CORENET system has received over 3.5 million online submissions of building plans. These massive amounts of electronic submissions have saved the construction industry a total of 40 million Singaporean dollars (equivalent of 29.5 million US dollars) in printing and shipping fees (Building and Construction Authority, 2013). Additionally, it saves paper, reduces unnecessary packaging waste and minimizes transportation induced pollutions. The CORENET system is an efficient, sustainable and eco-friendly way to process building permits, and is a leading example of why building permit automation is a rising necessity globally.

As a result of the CORENET system, Singapore was ranked number one in the world for the fastest issuance of building permits in 2012, as well as the world's most business-friendly country from 2010 to 2013 (Doing Business in a More Transparent World, 2012; Building and Construction Authority, 2013). According to the World Bank's Doing Business 2019 report, Singapore is the second fastest country to issue construction permits. The time it takes to process construction permits around the world are: 27.5 days in South Korea, 41 days in Singapore, 80.6 days in the United States and 117 days in Sweden (Doing Business Training for Reform, 2019). Singapore's status as a world-class builder is strongly associated with the successful implementation of the CORENET system. Automating construction compliance checking as an one-stop shop creates a business-friendly economy both beneficial to building professionals and business investors. Building permit automation is becoming an important topic internationally due to its contribution to a country's economic development.

2.1.2 South Korea

Another country that is leading in building permit automation is South Korea. As mentioned in the previous section, South Korea ranks no.1 in the world for the fastest country to issue construction permits (Doing Business Training for Reform, 2019). This ranking is partially due to the Architectural Institute of Korea (AIK), which was established in 1945. AIK is an organization that collaborates with government agencies and private firms to share and enhance architectural knowledge. Members of AIK are active in academic researches, conferences, seminars and workshops, with the aim of improving architectural standards in South Korea (Architectural Institute of Korea, 2016). AIK published the Korean Building Codes, which is uniform standard codes that regulate building designs and standards (National Building Codes and Standards of Korea, 2019). Korean building codes can be used by both architectural professionals and non-professionals. These codes are constantly updated and republished with accordance to changes in the architectural and constructional industries (Architectural Institute of Korea, 2016). The implementation of a national building code system creates a convenient environment for professionals and researchers in the building industry.

2.1.3 Other Countries

Another country that tries to optimize the construction permission process is India. India reformed the construction industry in 2017 by speeding up building permit processing time and reducing cost. Before the reformation, there were 37 procedures in Mumbai and 26 in Delhi. Now the number of procedures is reduced to 20 in Mumbai and 16 in Delhi. The time frame is also reduced from 128 days to 99 days in Mumbai and from 157 days to 91 days in Delhi (Ease of Doing Business Reforms 2014-18, 2018). So far, the reformation only applies to Delhi and Mumbai, but with the significant improvement in optimizing the building permission process, this reformation may be adopted by other cities in India.

India's neighbor China is also preparing for a reformation for building constructions and urban infrastructures. The number of days it takes for obtaining a construction permit in Shanghai is 169.5 days (Doing Business Training for Reform, 2019). With the new reformation, China aims to reduce it to 120 days (State Council reforms approval system of construction projects, 2019). By 2020, China aims to introduce a national construction and management system that examines and approves building proposals. The government of China hopes to both streamline the construction permit process and discourage illegal construction of buildings. The new system will simplify the documentation process, reduce wait time and provide public access to information. Not only will it create a convenient and centralized system for professionals, it will also create a safer living environment for all residents.

There are many other countries that optimize their construction permission process because an easier process opens more opportunities for investments and businesses. A country with easier and faster building permit process is more business friendly and more attractive for investors (Doing Business Training for Reform, 2019). Thus, building permit optimization has become increasingly important for major economic hubs like Singapore and developing countries like India.

2.1.4 Situations in Sweden

In Sweden, construction and development are regulated under the Planning and Building Act(2010:900) which contains the following provisions: 1) the planning of water and land areas, 2) issues with consideration of public and private interests, 3) requirement of a municipality to have detailed development plans that regulate construction and usage of the built environment. Other provisions include regional planning, requirements for construction, construction permit, compensation for damage, etc. The Planning and Building Act regulates construction, planning and development in a broad way to ensure a safe and sustainable living environment. It also serves as a guidance for municipalities' detailed development plans, which are more specific.

Under the Planning and Building Act, municipalities are responsible for regulating the usage and design of public space, business districts and water. Specific requirements embedded in the detailed development plans include: building height, footprint area, roof angle, distance to road, distance to boundaries, distance between buildings, etc. The municipality creates its own detailed development plans which may be adopted by the general region (Planning Process, 2018). A standard detailed development plan includes the following details:

- Areas where buildings are allowed
- Areas where buildings are not allowed (for example, borders between property and road)
- Green areas (parks)
- Roads
- Water bodies

Boverket, the National Board of Housing, Building and Planning, will require that all detailed development plans are in digital format and have uniform definitions (Planning process, 2018). This is important for the digitization of the surrounding areas as well as the adaptation of the detailed development plans by another municipality or region. With the ongoing housing crisis in Sweden and its long wait time for building permits (117 days), building permit automation has become increasingly important.

In Sweden, the building permit process can be broken down into 6 steps: 1) Inquire if a building permit is necessary through the local building committee. 2) Submit a written building permit application that contains drawings and other mandatory documents. 3) The building committee briefly reviews the application by checking the necessary documents and evaluating the quality of the drawings. 4) The building committee issues a notification of receipt if application is complete. 5) The building committee reviews the building permit application thoroughly to check if the measure meets the requirements. A decision should be made within ten weeks after the application is deemed complete. This study automates this step of the building permit process with the hopes of reducing processing time. 6) If a building permit is granted, neighbors and other stakeholders must be informed. Neighbors can object to the granted permit which can affect processing time, this adds uncertainty to the amount of time saved by automation. If a building permit is denied, appeal should be sent to the building committee in a reasonable time. The building permit becomes invalid if construction does not begin within 2 years, or if construction does not finish within 5 years, from the date of building permit issuance (Guide Permits and Notifications, 2018).

2.2 Previous Studies

Since building permission compliance checking is a complex process, optimization of the process has become an important topic internationally. Studies related to building permit automation range from translation of rule sentences to compliance checking of specific building requirements, such as building envelope design, building height and footprint areas (Tan et al. 2010; Olsson et al. 2018).

2.2.1 Digitization of the Built Environment

The digitization of buildings is the key to understanding the relationship between a building and its nearby environment. Digital data is collected and distributed across cloud services, with computers that operate 24 hours a day (Whyte and Hartmann, 2017). Building digitization and modeling also allows building information to be reused, monitored and updated throughout the planning process. Building digitization is heavily promoted in international organizations and governments. Various studies reveal that ongoing digitization influences organizational structure and process. Whyte and Hartmann (2017) showed that digitization can streamline processes while Papadonikolaki and Wamelink (2017) showed that digitization create a simpler connection between supply chains within and across firms. However, implementing Building Information Model (BIM) technologies on a project is a complex process, it requires continuous adjustments and adaptations (Whyte and Hartmann, 2017). BIM allows building data to be assessed for planning, such as the design and construction of building related infrastructures. It is therefore crucial to embrace flexibility on the adaptation of BIM on building projects and project supply chains.

2.2.2 BIM data to geospatial data

Upon digitization of the surrounding area, another initial step of building permit automation is the integration of building information models and geospatial information, which includes the conversion of BIM data to geospatial data, such as CityGML. CityGML stores 3D models of cities and landscapes and defines the relationship between different objects in a city, such as roads, vegetation, building, etc. The conversion of BIM data to geospatial data allows users to perform geospatial analysis on 3D data, which in turn enables 3D checking of building permits.

The most commonly used data format in BIM is Industrial Foundation Classes (IFC), which contains both geometric and semantic building information. These information cannot be accessed by GIS software due to data incompatibility. In order to access the building information, transformation from IFC to geospatial format is required. Zhu et al. (2018) conducted a study on IFC to shapefile transformation and concluded that direct transformation preserves the most information and geometry. Stouffs et al. (2018) automated the conversion of IFC to CityGML while retaining both geometric and semantic information. Although it is difficult to achieve complete transformation without any loss of data, the use of Application Domain Extension (ADE) and Triple Graph Grammar (TGG) have proven to be effective (Stouffs et al. 2018). CityGML has 5 levels of detail (LOD), most IFC to CityGML models have LOD1 or LOD2, which is coarse and lacks detail information. Higher level of detail is more ideal because it contains useful architectural details such as doors and windows, but is challenging to achieve. Donkers et al. (2015) investigated this issue and successfully developed an algorithm that converts IFC to CityGML in LOD3. The transformation of IFC to geospatial shapefile or CityGML is an important step to analyze building information in a GIS software.

Furthermore, Van Berlo et al. (2013) showed that it is possible to integrate 3D designs with geospatial domains for 3D building checking. This is achieved through three steps: 1) conversion of 2D spatial planning maps to 3D, 2) integration of spatial plans with IFC data, 3) compliance checking of building regulations from integrated data. However, integration is not always successful and can be very complicated. Arroyo Ohori et al. (2018) showed that the integration of BIM data and geospatial data is a complex process that contains errors and problems. One major problem is the differences in geometry, topology and semantic information between a BIM and geospatial model (Floros et al. 2017). Due to different standards, geometries and topographies that are supported in BIM might not be supported in a GIS software. So geometrical or topological errors in a BIM must be corrected for successful conversion (Arroyo Ohori et al. 2018). Another common issue is the various interpretations of IFC to CityGML transformations, the different interpretations is due to the fact that IFC classes must be condensed into a few CityGML classes. Subsequently, details from BIM data are simplified and generalized when converted to GIS (Arroyo Ohori et al. 2018). Although there are successful BIM to GIS conversions, the process is complex and it involves various steps of error corrections before successful conversion.

2.2.3 Translation of Building Codes

The building permit automation process involves different steps, one of which is the transformation of legislations from paper format to digital format. Previous studies have shown different ways to convert rule sentences into computational building codes. Lee et al.

(2016) converted sentences of regulations from the Korean Building Act into computer readable forms. 15,000 building permit-related sentences are stored in a database as intermediate codes ready for the rule checking process. Additionally, Kim et al. (2019) developed a rule translation method that converts natural languages to code using parametric input tables and programming languages. Results are represented in easily readable code called KBimCode, which is simple and intuitive because it uses symbols rather than programming codes.

Another example of building code compliance checking is a study by Macit İlal and Günaydın (2017), who developed a method called RASE (Requirement, Applicability, Selection, Exception) that deconstructs rule sentences semantically into four groups: 1) Requirement – define the conditions to be fulfilled by one or more building features, 2) Applicability – determine to which building feature the requirements apply to, 3) Selection – identify conditions if the rules apply only for specific features, 4) Exception – identify conditions if rules do not apply to any building features. The RASE deconstructs rule sentences and represents building codes in a simple way, which can be easily interpreted by non-programmers. Researchers have successfully developed ways to translate written rule sentences into digital formats, but the process is complex because it is complicated to decompose natural languages into computer-readable building codes.

2.2.4 Building Envelope Design Automation

After rule sentences are converted to digital building codes, the digital data are used to validate different aspects of a building. Several studies have been conducted on building compliance checking, for example, Tan et al. (2010) presented new ideas to automate the checking of building envelope designs. Building envelope is the separation between conditioned and unconditioned environment, such as water, heat and light. The approach starts with an extended building information model (EBIM) that allows software to extract data as input and store the results in the EBIM. Rule sentences are presented in decision tables for a clear representation of complex logic. In order to check building envelope designs against the regulations, an extended building code (EBC) is used because it can handle XML-based decision tables and building codes. The ideas proposed by Tan et al. (2010) allows regulations in the EBC decision tables to be updated without making changes to the rule checking engine. The proposed approach integrates hydrothermal simulation of the building envelope, EBIM and EBC decision tables. The result is a report that assesses the compliance checking results to give advice for identifying and fixing design errors.

2.2.5 Building Height and Footprint Area Automation

Besides building envelope design, other commonly tested building requirements are: building height, footprint area, elevation, roof angle, etc. Olsson et al. (2018) conducted a study to automate the checking of building height and footprint area in accordance to the detailed development plans in Sweden. Building height is calculated by transforming BIM data to CityGML, integrating the transformation with ground elevation, as well as evaluating the ground plane, facade plane and 45-degree plane of a building. Two height methods are used and both proved that it is feasible to automate the calculation of building height. Building area is calculated by using various IFC elements to visualize and quantify the base area of the building objects. Eight buildings with different levels of detail are tested and all have shown reliable results, with less than 1% difference from manually measured area. The study proves that it is possible to automate building requirement compliance checking based on regulations listed in the detailed development plans, which may help increase the efficiency of the building permit process.

3. Methodology

3.1 Legal Perspectives

3.1.1 Building Placement Regulations

According to the Building and Planning Act, a building permit is required for new construction and extensions. Alterations to a building also requires a permit if the alteration involves: 1) changing the usage of the building, 2) creating premises for non-residential purposes, 3) changing the appearance of a building, such as color, facing and roofing materials.

A building permit is not required for: separate outbuildings, garages, erecting or extending balcony, bay window and minor extension (less than 15 square meters gross floor area) if the proposed changes do not occur within 4.5 meters from the site boundaries. This regulation applies to other building properties as well, a full list can be found in the Planning and Building Act. Some building properties may occur closer than 4.5 meters from the site boundaries if the neighborhood allows it (Planning and Building Act (2010:900) 2010).

3.1.2 Parking Regulations

According to the detailed development plans from Höganäs municipality, the parking space requirement is 5m x 2.5m. In this thesis, all parking space must be directly connected to the road and must be perpendicular to the road. Street parking is not considered in this case.

3.2 Process Perspectives

3.2.1 Building Placement Parameter

Although the required minimum distance allowed for a building is 4.5 meters from the boundaries. There are exceptions if the neighborhood allows it. Therefore, it is important to check for the closest possible minimum distance for a building in the dataset before applying any test. In this case, the Planning and Building Act is only used as a guideline, the actual allowed minimum distance for construction is based on the requirements in the detailed development plan.

3.2.2 Parking Complications

Parking availability is a topic that is less explored in building permit automation, but is nonetheless an important aspect of the building compliance checking process. The algorithm developed in this study checks for potential parking availability and retrieves areas that are permissible for parking. These potential parking areas are dependent on the size, orientation and placement of the house. For example, if the house is too large or too close to the road, then potential parking space is limited. Since parking availability is dependent on the placement of the house, there are multiple factors that must be taken into consideration. These factors include the shape of the property and house, the number of lines that intersect the property and the road, as well as the placement of the house. These factors altogether create numerous scenarios that can cause complications to finding parking availability. Figure 1 indicates some of these scenarios that may occur for finding parking availability. There are numerous scenarios that may cause complicated geometrical issues. It is not feasible to simply create a 2.5m x 5m rectangle to test for parking availability because it is difficult to determine the placement of this 2.5 x 5m polygon. A suitable approach is to visualize potential parkable space. The developed algorithm is able to solve geometrical issues and determine parking availability from the placement of the house, regardless of its orientation and proximity to the road.

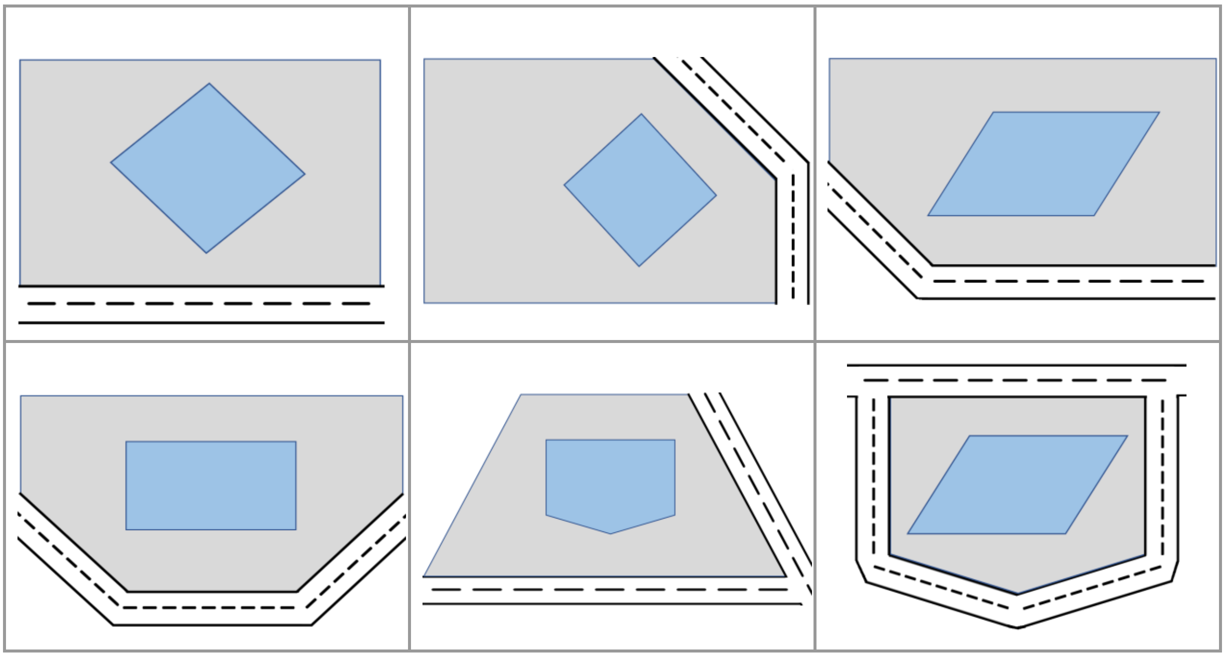


Figure 1. Potential parking scenarios. Blue polygon: building; grey area: property; dashed lines: roads. Parking space must be directly connected to a road.

3.2.3 Program Logistics

The algorithm developed in this study performs compliance checking of all four building requirements independently. Rather than checking one requirement at a time, the algorithm checks all building requirements at the same time. The results of one building property is independent from another building property. Checking all building requirements at the same time enables the user to correctly identify which building properties are according to regulations and vice versa.

3.3 Building Height and Roof Angle

Building height and roof angles are two commonly tested building requirements. The building height and roof angles of the house are compared to the regulations defined for the neighborhood. The first step is to rotate and offset the house to a desired location, then the house and neighborhood layers are overlaid to retrieve and compare attribute values. The requirements for building height and roof angles are stored in the neighborhood layer where buildings are allowed (the yellow area shown in Figure 2), so it is important that the house location intersects with the correct layer. The program displays a message that indicates the results of the test. The program proceeds to check for building placement regardless of the results.

3.4 Building Placement

3.4.1 Check Building Location

The first step in the house placement part of the algorithm is to check whether the house is located within an area where buildings are allowed. It is not enough that the house is inside a buildable neighborhood, the house must also be completely within a property area, as shown in Figure 2. The minimum distance requirement for building includes the pink area where buildings are forbidden. It is therefore essential to extract two polygons, one with the non-buildable area and one without. A spatial filter is applied to the polygon without non-buildable area to check whether the house is within an area where buildings are allowed. For the program to pass the transformation, the house must be completely within a property, any intersection will result in a fail. If the house location is permissible, the program will proceed to the next step to calculate minimum distance for the polygon with non-buildable areas. If house location is not permissible, then transformation is terminated.

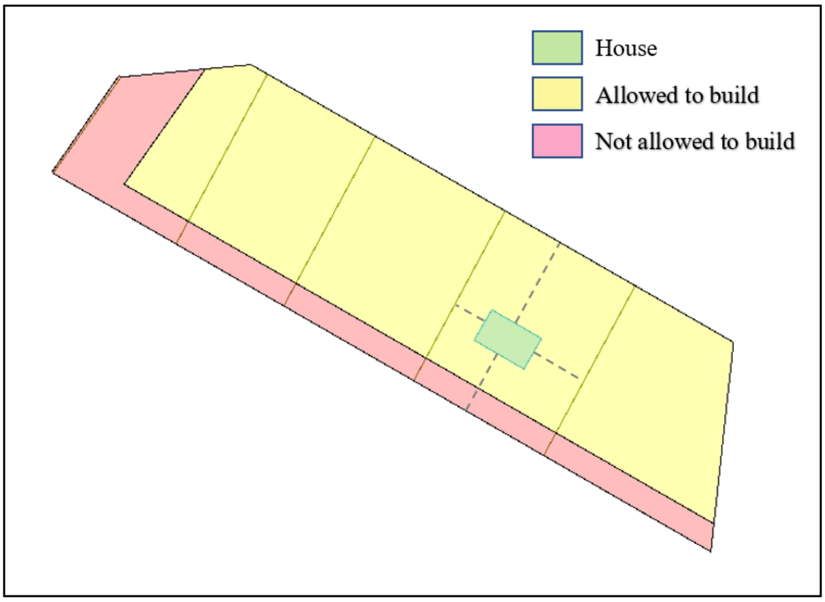


Figure 2. Building position and area layout.

3.4.2 Check Minimum Distance

There are different ways to check if the location of the house is within the legal minimum distance from the boundaries, as required by the detailed development plans. One way is to retrieve the minimum distance between two polygons, in this case the two polygons being tested are the house polygon and the area polygon. Once the minimum distance value is retrieved, this value is compared to the legal minimum distance for building. With this approach, only one side of the polygon is checked to determine whether or not all four sides of the polygons are according to standard. This concept is explained in Figure 3 and Figure 4: the big rectangle is a property boundary, the grey area illustrates the minimum distance requirement for the property and the small green rectangle is a house.

In Figure 3, the minimum distance allowed for a building according to the detailed development plan is 6 meters, and the actual minimum distance measured between the house boundary and the property boundary is 5 meters. Since the shortest distance (5m) is shorter than the required minimum distance (6m), the location of the house is too close to the boundary and does not conform to the regulations. The program will in turn reject the transformation. In this case, it does not matter that the other 3 sides of the house are within the allowed distance, if one side fails, the house location will be rejected.

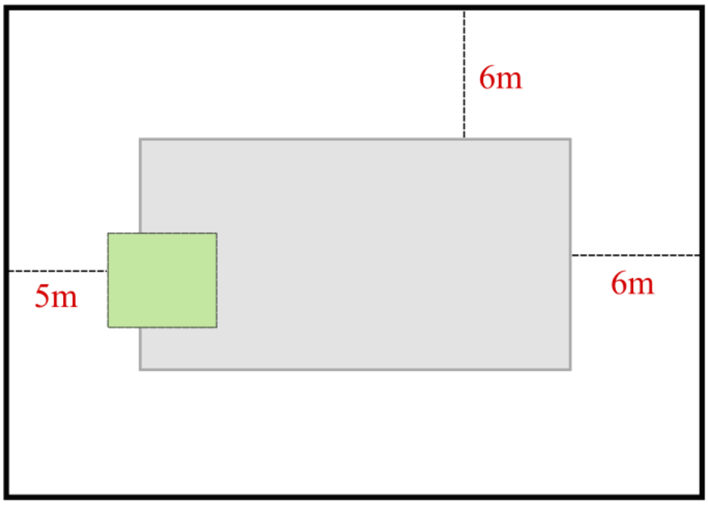


Figure 3. Illustration of using minimum distance calculation to determine if a building is inside a buildable area (grey polygon).

In Figure 4, the minimum distance allowed for building according to the detailed development plans is 4.5 meters, and the actual minimum distance measured between the house boundary and the property boundary is 5 meters. If the side of the house with the shortest distance (5m) is greater than the required minimum distance (4.5m), all sides of the house must also have distance greater than 3m. In this case, all sides of the house are within the allowed distance and the program will therefore accept the house location.

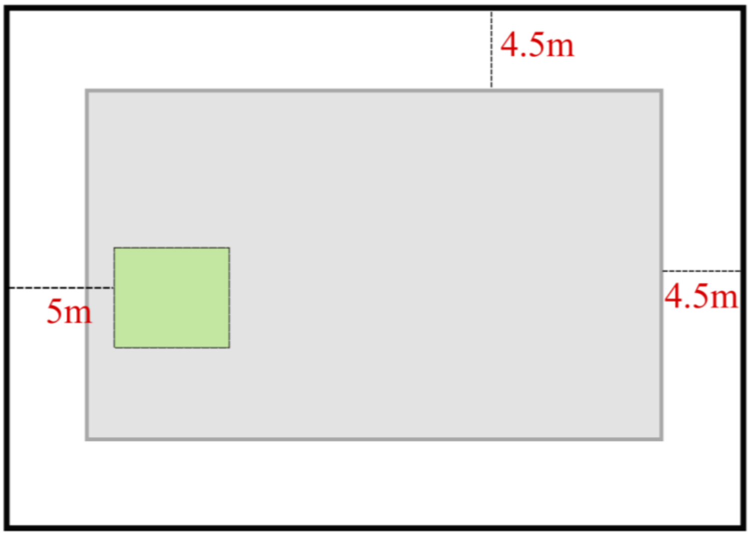


Figure 4. Illustration of using minimum distance calculation to determine if a building is inside a buildable area (grey polygon).

3.4.3 Identify Lines with Minimum Distance Requirement

If the house location is permissible but minimum distance is not, then the program displays the line(s) that the house is too close to, as well as the distance that is intruding. This is done by converting the property boundary from polygon to separate lines. Then a distance check is applied to all of those lines with the maximum distance being the minimum distance requirement. For example, if the property polygon has 4 lines and a minimum distance requirement of 8m, the program runs 4 tests to compare all 4 lines to the house object to retrieve the distance. Any line that is further than 8m from the house passes the test and is colored green, all the other lines fail and are colored red. Then, the algorithm creates a buffer based on the user-input minimum distance requirement around the red lines. These buffers create a visual representation of how far the house is from legal regulations. In the meantime, the program computes the difference in distance between the current house location and the legally allowed minimum distance.

3.5 Parking Availability

3.5.1 Discover Potential Parking Area

The initial step of finding parking availability is to identify potential areas that are suitable for parking inside the property. Since one of the requirements for parking is its connection to a road, it is important to treat the property and road data as both lines and areas for different transformations. The program converts property area and road data to lines to retrieve the line of intersection.

From the retrieved line of intersection, the algorithm creates a 5m buffer. The 5m buffer is based on the requirement that parking space from the road must be at least 5 meters and perpendicular to the road. This 5m buffered area is potential parking space. It is important to clip the buffer with the property to make sure the 5m buffer does not extend outside of the property. Since the buffer can only extend to the property, it may result in different shapes depending on the angle of the property's corners. Buffers might result in rectangles, tetragons or trapezoids. Tetragons and trapezoids are created if the corner that intersects the road has an angle less than 90 degrees. If there is one corner with an angle less than 90 degrees, the 5m buffer becomes a tetragon. If there are two corners with an angle less than 90 degrees, the buffer becomes a trapezoid. Figure 5 displays how the line of intersection between the property and the road is buffered and clipped. For a simple demonstration, the examples in Figure 5 only have one line of intersection between the property and the road. However, the algorithm is capable to process multiple lines of intersection.

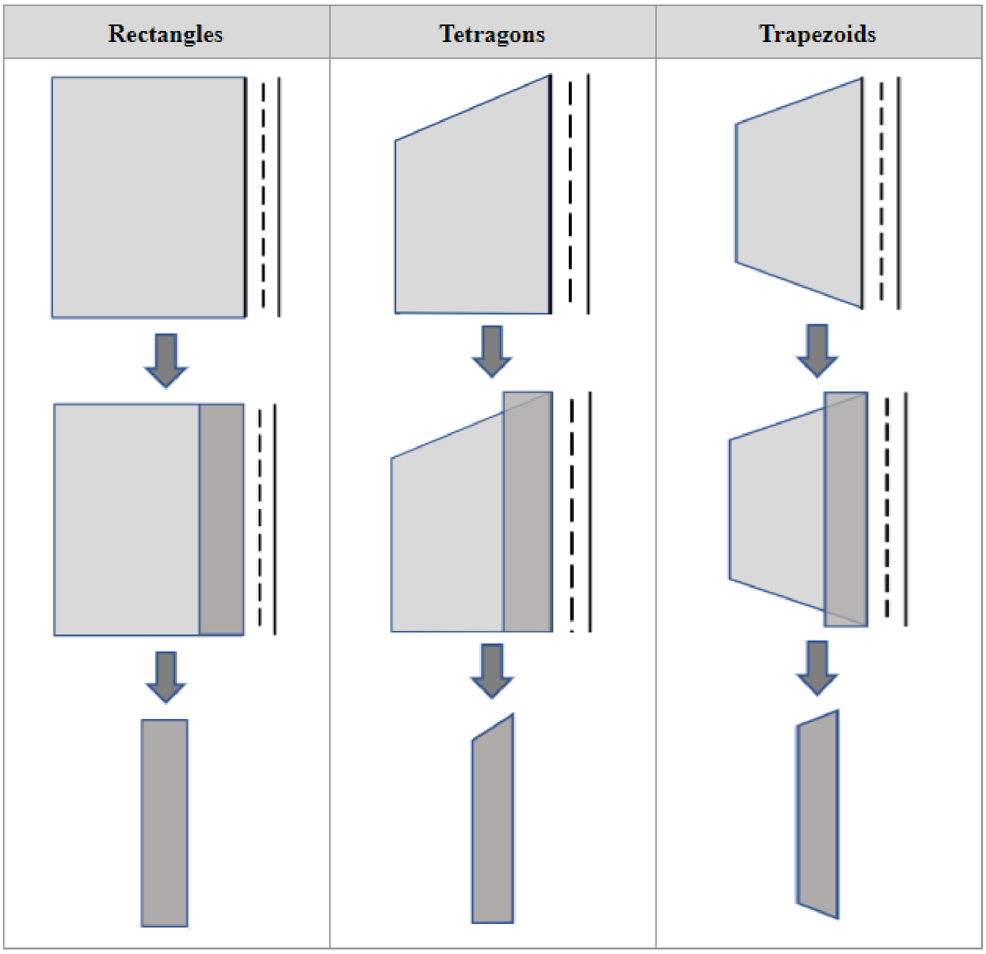


Figure 5. Demonstration of creating a 5m buffer from the line of intersection between the property and the road. The buffer is clipped using the property.

3.5.2 Multiple Intersected Lines

In a more complex case with multiple lines of intersection between the property and the road, it is important to treat those lines individually by adding an unique ID. The buffered polygons should have the same unique ID as the line that intersects them. This step is important because FME runs transformers to all polygons at the same time, instead of applying transformers to one polygon at a time. By giving the polygons and lines a corresponding unique ID, the program is able to use the group-by function to process one polygon at a time. A custom transformer is recommended in this case, which is the compression of a series of transformers into a single transformer.

3.5.3 Polygon Transformation

One of the parking requirements states that parking must be perpendicular to the road, this means that the edges of a tetragon and trapezoid are not parkable. The algorithm transforms any trapezoids or tetragons into rectangles to eliminate unparkable edges. This is done by retrieving the vertices of the polygon, and finding the shortest distance between the vertices and the line of intersection between the property and road. The outputs are then used to create line connections between the vertices and the line of intersection. In a mathematical sense, the shortest distance between a point and a line is always perpendicular to the line. The algorithm uses the newly formed perpendicular lines to cut the tetragon/trapezoid into segments. The program checks the geometry of all these segments to filter out triangles; all triangles are colored red and are considered unsuitable for parking.

Since parking must be perpendicular to the road, the polygon transformation method is applied to all polygons regardless of its geometry. This is because what appears to be a rectangle may not always be a rectangle, sometimes these small differences are enough to hinder the results. The algorithm is able to identify small differences and eliminate them, so the outputs are always rectangles. If the input polygon is indeed a rectangle, then there is no change. If the input polygon is a tetragon, the resulting segments are a rectangle and a triangle. If the input polygon is a trapezoid, the resulting segments are a rectangle and two triangles. The tetragon/trapezoid to rectangle transformation process is demonstrated in Figure 6. Any triangles that result from the transformation are considered unsuitable for parking and are colored red. Any rectangles that result from the transformation are potential parking space and will be tested in the next step.

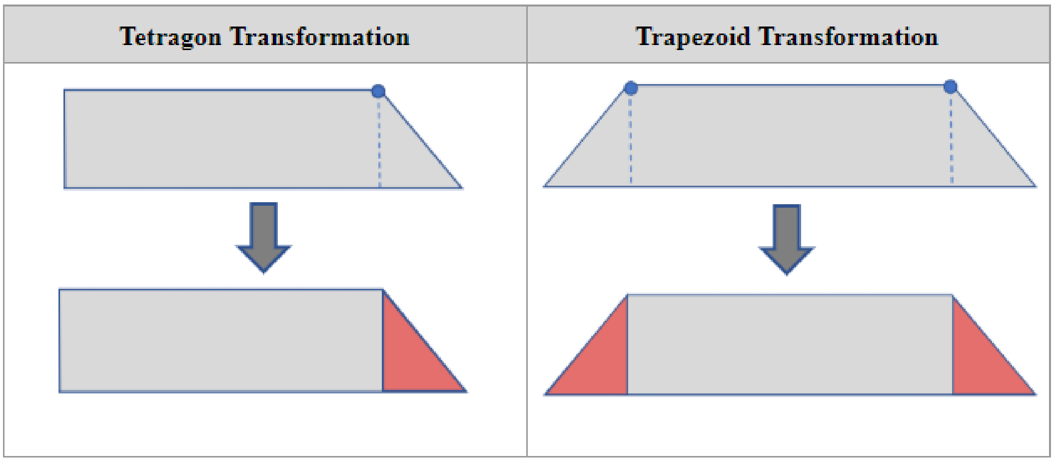


Figure 6. Tetragon/trapezoid to rectangle transformation.

3.5.4 Intersections - No Intersection Between House and Rectangles

Although the resulting polygons from the transformation are all rectangles, whether or not they are suitable for parking depends on their width/length. Before checking the width/length, it is important to first check if the house intersects with any of these rectangles. If the house does not intersect with any of these rectangles, the algorithm calculates the width/length of the rectangles. Since all transformed polygons are rectangles and the buffer is 5m, the width/length can be calculated through equation 1: $(p - 2*h)/2 = l$ where p is perimeter, h is height and l is length. For example, if the perimeter of the rectangle is 22, then (22-10)/2 = 6m. Since 6m is greater than 2.5m, there is sufficient space for parking and the rectangle is colored green, as shown in Figure 7. If the calculated value is less than 2.5m, then there is no sufficient space for parking and the rectangle is colored red. The same process is applied to all rectangles that are disjoint from the house.

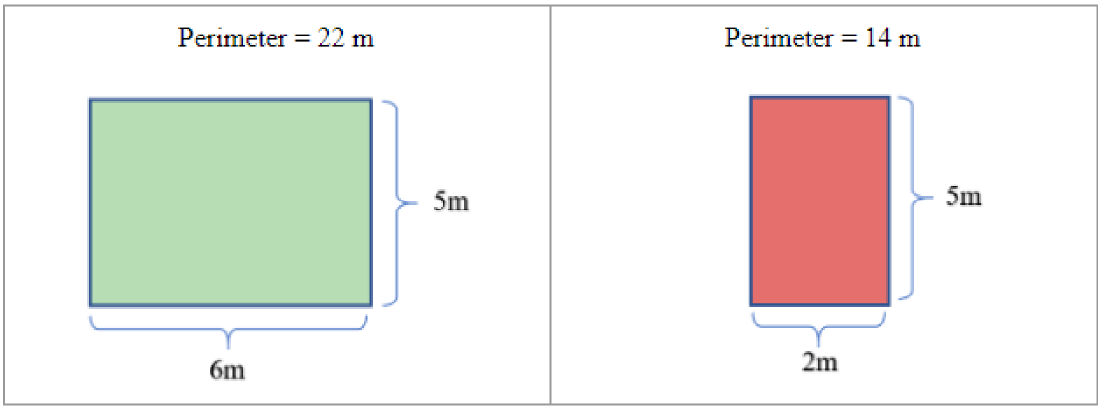


Figure 7. Rectangles width/length calculation based on perimeter.

3.5.5 Intersections - Intersection Between House and Rectangles

If the house intersects any of the rectangle, then potential parking space is reduced. The algorithm overlays the house with the rectangles to retrieve the area of overlap. Nodes are then retrieved from the overlapping area. For this step of the algorithm, it is important to take all nodes into consideration rather than only the nodes that touch the boundary, this is because a house can have different shapes and can intersect the rectangle in different ways. By applying the process to all nodes, the program handles all possible ways of intersection. For each node, the algorithm creates a perpendicular line to the line that intersects the property and the road. This method is similar to the one in Section 3.5.3, the major difference is that the perpendicular lines must be extended so they reach both boundaries. Since the buffer is 5m, it is safe to extend the lines by 5m, this is simpler than calculating the distance of extension for each line. The extended lines are then used to cut the rectangle into segments. The algorithm checks each of these segments for possible intersection with the house. All segments that intersect the house are merged together as one polygon and colored red. This process is demonstrated in Figure 8. The merged polygon is no longer suitable for parking because potential parking space is reduced by the house. For segments that do not intersect the house, the algorithm calculates their width/length using the methods mentioned in Section 3.5.4. Polygons with width/length greater than 2.5m are suitable for parking and are colored green. Polygons with width/length less than 2.5m are unsuitable for parking and are colored red.

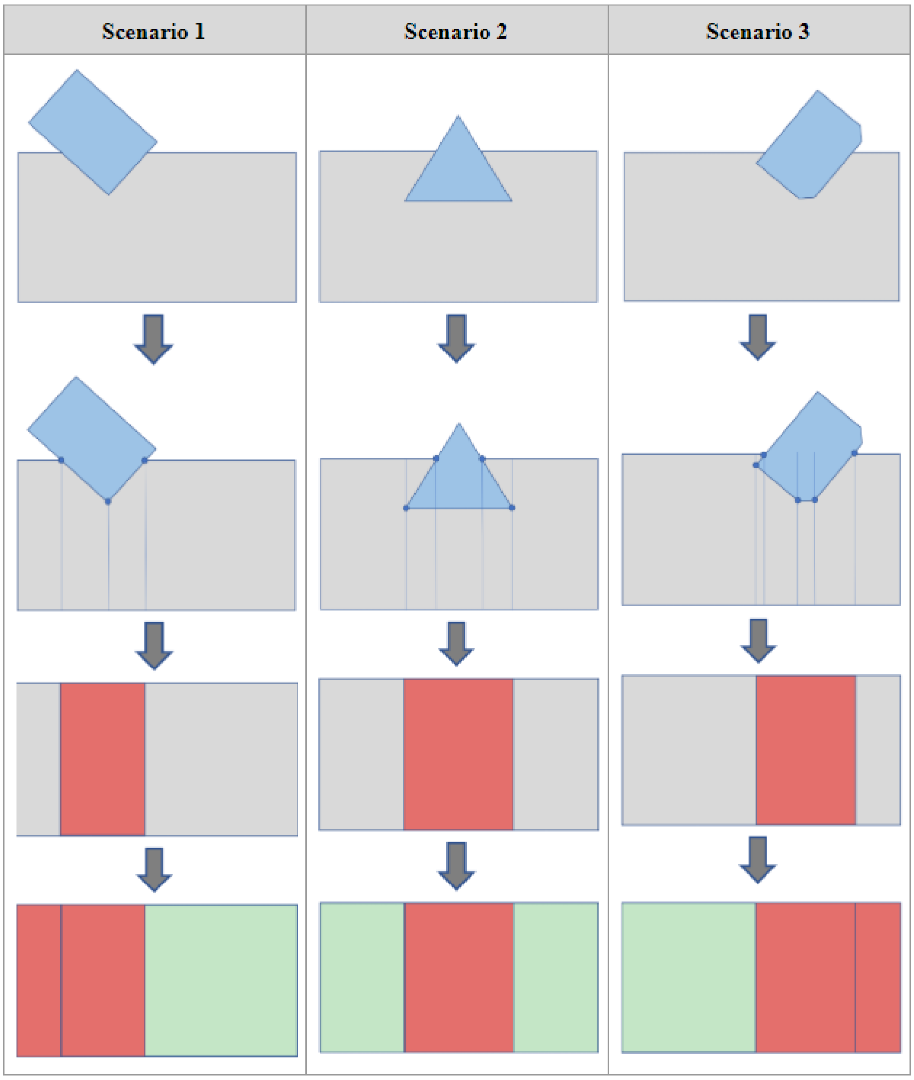


Figure 8. Determination of parkable and non-parkable space due to house intersection. Non-parkable space(red): intersection with house or insufficient parking space. Parkable space(green): no intersection and sufficient parking space.

4. Implementation

4.1 Software

The software used for this thesis is Feature Manipulation Engine (FME) (SAFE Software, Vancouver, Canada), a credible data integration platform that connects and transforms data between different systems. FME reads and transforms the data from the detailed development plans and building footprints to perform analysis.

4.2 Program workflow

4.2.1 Expected Building Placement Outcome

The algorithm checks whether the house location is according to the regulations, and if the house location meets the minimum distance requirement in the detailed development plans. In this study, users can alter the minimum distance requirement for testing purposes. While running the program, three types of results might occur:

1. If the house location is not according to the requirements, the transformation will fail and no further action will be taken. These locations include:
    a. If the house falls outside of buildable areas.
    b. If the house intersects any property boundaries.
    c. If the house intersects area that are not allowed for building.
2. If the house location is according to the requirements, the program will check whether or not it meets the minimum distance requirement. If it does not meet the requirement, the program will perform the following:
    a. Display a message that the house location does not meet the distance requirement.
    b. Color the lines that are within the minimum distance requirement green and color lines that do not meet the requirement red. A buffer is created using the minimum distance requirement.
    c. Calculate and display the difference between the measured distance and the required distance.
3. If the house location meets all conditions of the detailed development plans, the transformation will pass and all lines are colored green.

The program utilizes multiple transformations to achieve the results. Figure 9 illustrates the steps necessary to address the following: 1) Check requirements for house location. 2) Compare the measured minimum distance with the required minimum distance. 3) Retrieve distance to move and create a visual representation of results.

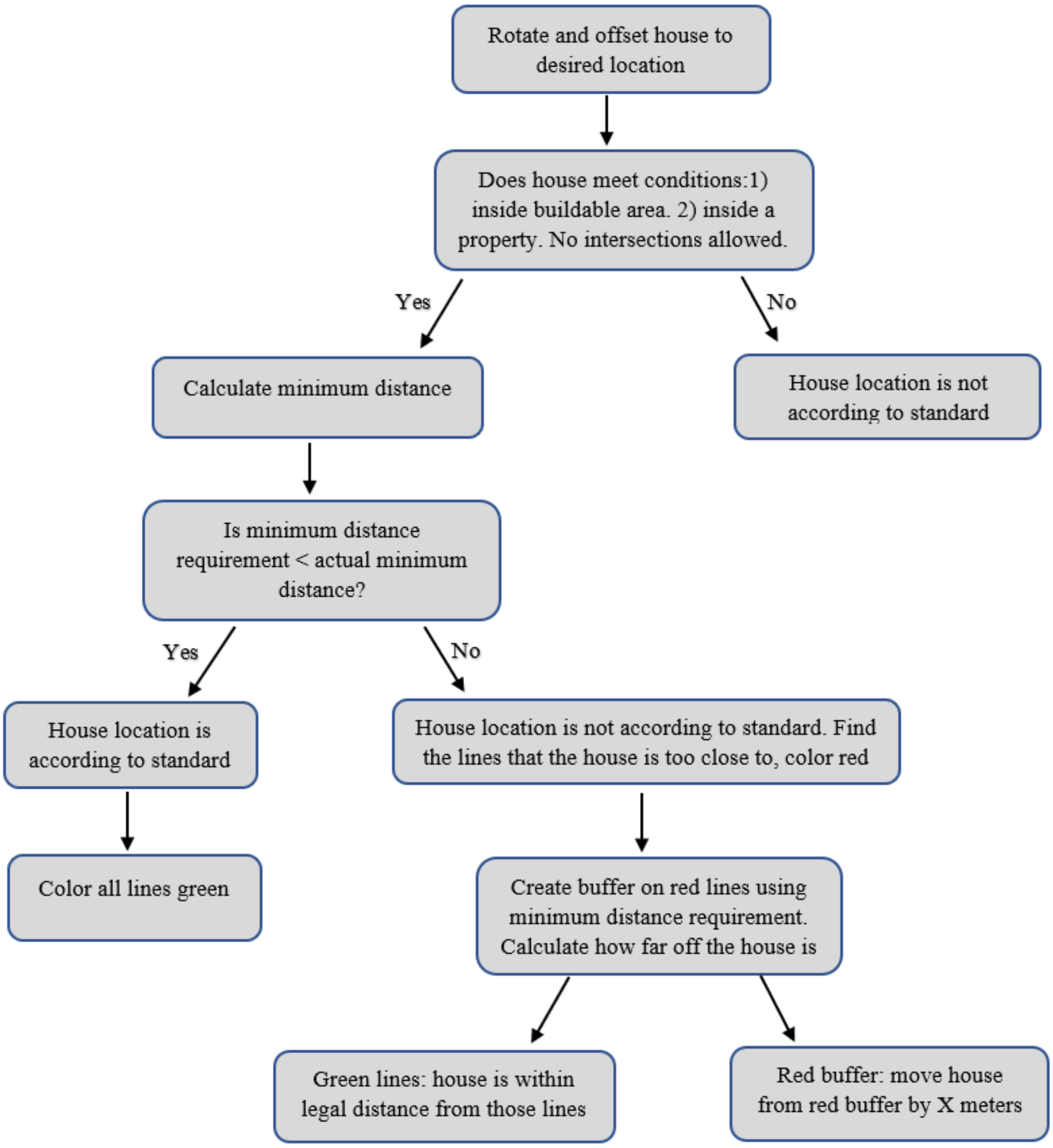


Figure 9. Flowchart of housing placement algorithm.

4.2.2 Expected Parking Availability Outcome

An essential functionality of the algorithm is its ability to determine parkable and non-parkable space. Parkable space is when the width/length of a potential parking area is greater than 2.5 meters. Non-parkable area refers to triangular edges, insufficient parking space and areas that are interfered by the building's location. There are 4 scenarios that can result in unsuitable parking space and only 2 scenarios that result in suitable parking space.

Figure 10 shows the general structure of the parking availability process. Red boxes in Figure 10 indicate that the outcomes are unfit for parking and green boxes indicate that the outcomes are fit for parking.

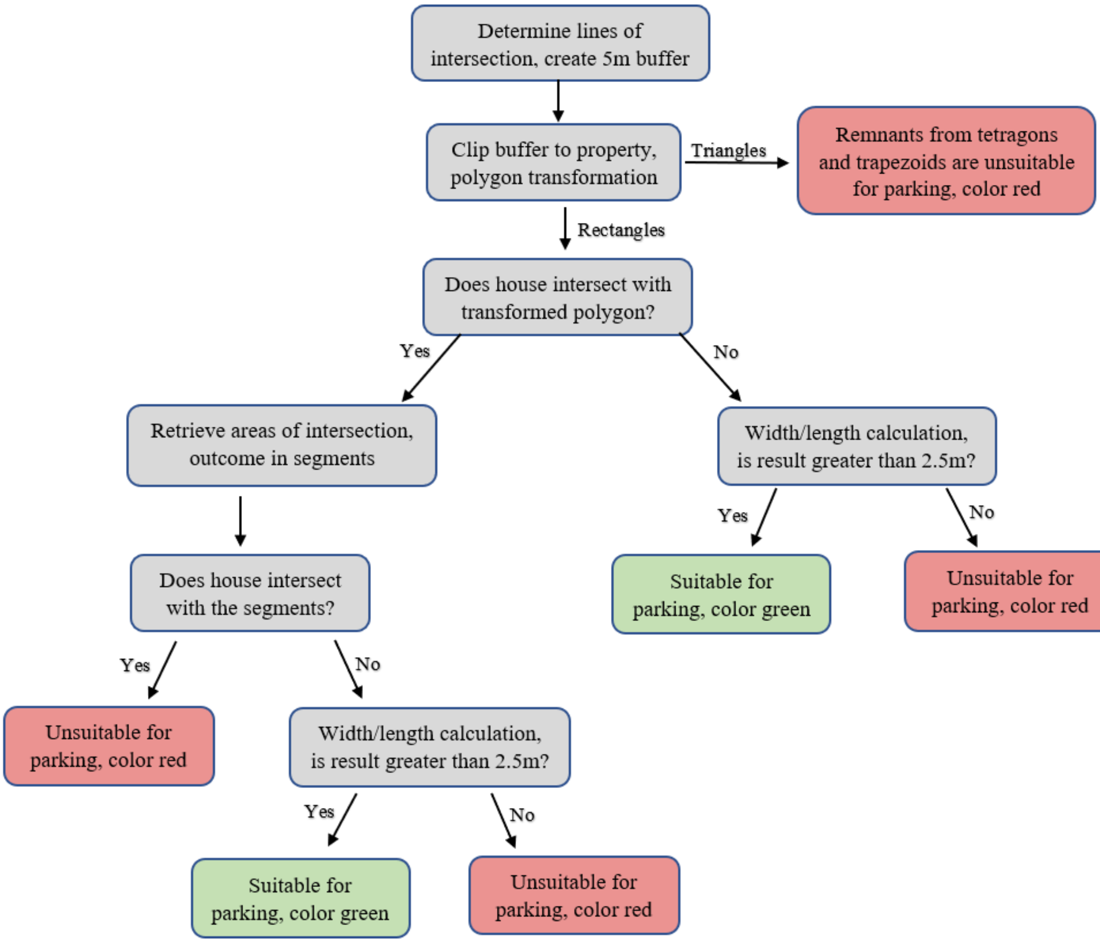


Figure 10. Flowchart for parking availability algorithm.

4.3 Licensing

The developed algorithm is published under BSD-3-Clause license and can be accessed in Github through the following link: https://github.com/TestbedLU/Testbed_BIM_GIS/tree/master/Building%20Location%20and%20Parking%20Availability

5. Case Study

5.1 Study Area

The detailed development plans used in this thesis are from Höganäs municipality in Southern Sweden. The detailed development plans are in digital format and according to the Swedish standard SS637040:2016. The following information is retrieved from the detailed development plan: area boundaries, minimum distance requirement and parking space regulations. These parameters are used for testing the algorithm and the developed algorithm should identify whether building features comply to regulations in the detailed development plans.

In this study, property refers to the piece of land where the house is located. Neighborhood refers to a land that has multiple properties. Some building requirements, such as building height and roof angle are neighborhood specific while parking regulation is municipality specific.

5.2 Data

The data used in this thesis are acquired from different authorities and institutes, all dataset are in reference system SWEREF99 13 30.

- ➢ Detailed development plan from Höganäs municipality
- ➢ Property boundaries
- ➢ House footprint

Although the required minimum distance allowed for a building is 4.5 meters from the boundaries. The Högänas detailed development plan allows for construction closer than 4.5 meters from the site boundaries. In some areas, construction is allowed as close as 3 meters from the boundaries. As a result, tests performed in this study use different distance parameter but not closer than 3 meters, to show that the developed algorithm can perform calculation regardless of the distance parameter.

5.3 Data Manipulation

The data used in this study is produced by different authorities and institutes, which can create discrepancies between the data. These discrepancies can hinder the results of the algorithm and yield incorrect results. Therefore, it is necessary to correct the data prior to the start of the algorithm. This section describes the different types of data correction methods applied specifically for this dataset, so they might not be applicable for another set of data.

5.3.1 Parameters

The detailed development plans used for this study do not have the attribute values for the minimum distance required for building. So the first step is to add the attribute values and a user parameter. Adding a parameter to the transformer makes the program more user friendly. The minimum distance requirement can be altered freely for testing purposes, this is because the minimum distance requirement for one area might differ from another due to structural differences. The unit of the attribute values is in meters.

5.3.2 Reference System Check

When overlaying the detailed development plan data with the property boundaries data, it is very important that both dataset are in the same reference system. Using different reference system leads to disaligment of layers which can hinder the algorithm. In this thesis, the detailed development plan is reprojected from SWEREF99 18 00 to SWEREF99 13 30 in order to match the local reference system for Höganäs municipality.

5.3.3 Rotation

Since properties vary in sizes, sometimes a building object must be oriented in order to fit inside a property. The house polygon in this study is not at origin, so it is important to reset the origins to (0,0) for two reasons because applying a rotation at (0,0) is easier and more intuitive than applying rotation at any other point.

5.3.4 Gaps and Overlaps

In this study, the property boundaries dataset is not digitized according to the detailed development plan, as there are many gaps and overlaps between the property boundaries and the roads. Some are small (0.04 cm) and some are big (17 cm). Before running spatial analysis, these gaps and overlaps must be eliminated. The program snaps lines with a gap or overlap of less than 20 cm to the nearest line segment. The 20 cm parameter is only applicable to this study and might not be applicable for another dataset. This thesis highly recommends property boundary dataset to be digitized using a snapping tool so the boundaries are aligned with the same boundaries in the detailed development plan.

5.4 Results

5.4.1 Building Placement Results

To test the functionality and accuracy of the program, a set of nine tests are performed under reference system: SWEREF99 13 30. To ensure that the program can accurately calculate and compare the required distance with measured distance, location and/or distance requirement varies for each test. Table 1 shows the inputs and results of all nine tests. The results of table 1 are shown in Figure 11 with corresponding ID numbers. If the house location does not conform to standards, the program is unable to measure any distance, as shown in Table 1 and Figure 11 ID 1, ID 2 and ID 3. If the house location meets the minimum distance requirement, no information is displayed for distance to move, as shown in ID 4 and ID 7. If the house location does not meet the minimum distance requirement, the program displays the distance to move in order for transformation to pass, as shown in ID 5, ID 6, ID 8 and ID 9. The lowest chosen distance parameter is 3 meters because the neighborhood allows for construction as close as 3 meters from the site boundaries, as opposed to 4.5 meters in the Planning and Building Act.

Table 1. Nine tests of building placement with varying distance requirement and coordinates.

ID	X-coord	Y-coord	Distance requirement*	Measured distance*	Distance to move*	Expected outcome	Program outcome
1	102025	6234170	5	N/A	N/A	Fail	Fail
2	102076	6234064	3	N/A	N/A	Fail	Fail
3	102090	6234185	3	N/A	N/A	Fail	Fail
4	102091	6234190	5	8.13	N/A	Pass	Pass
5	102091	6234190	9	8.13	0.87	Fail	Fail
6	102091	6234190	10	8.13 9.49	1.87 0.51	Fail	Fail
7	102124	6234072	3	3.41	N/A	Pass	Pass
8	102123	6234072	3	1.59	1.41	Fail	Fail
9	102123	6234072	5	1.59 4.51	3.41 0.49	Fail	Fail

**All distances are in meters (m).*

Accuracy: 100%

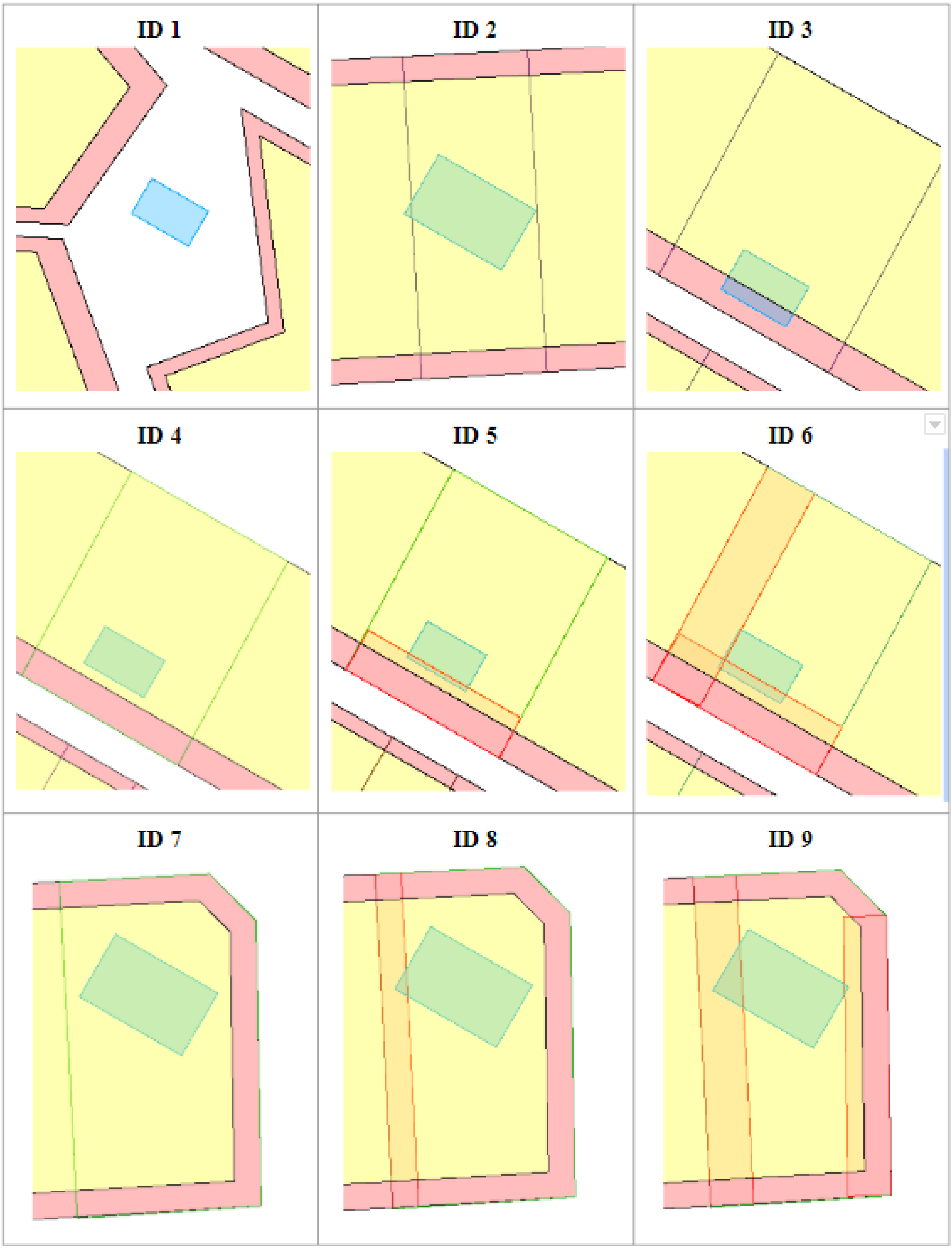


Figure 11. Building placement testing results. ID 1-3: building is in a non-buildable area. ID 5, 6, 8 and 9: building is too close to property boundaries, red buffer represents minimum distance requirement. ID 4 and 7: building meets minimum distance requirement.

5.4.2 Parking Availability Results

5.4.2.1 Simple Case - One Line of Intersection

This section shows the results for properties with only one intersected line or two non-consecutive intersected lines between the property and the road, as shown in Figure 12. Polygons that are colored green means that they are rectangles and have a width greater than 2.5m. Polygons that are red means they are unsuitable for parking for the following reasons:

- Polygons are non-rectangles.
- Polygons intersects the house.
- Polygons with less than 2.5m width/length.

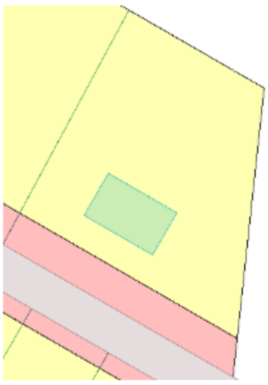
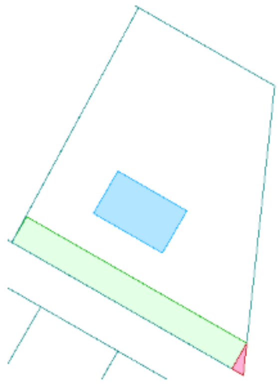
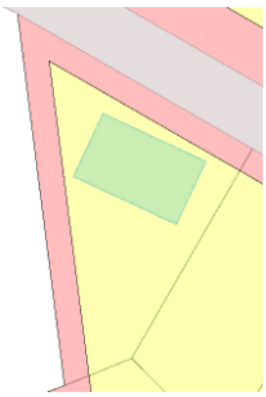
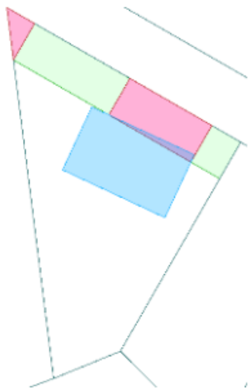
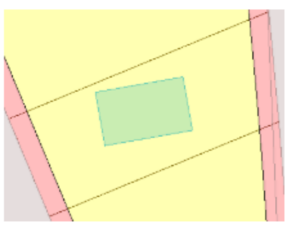
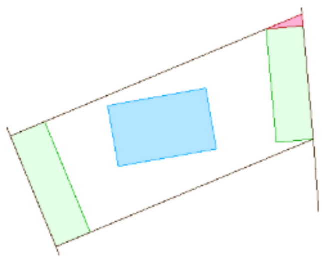
ID	Coordinates	House location	Parking Availability
1	X: 102123 Y: 6234175		
2	X: 102054 Y: 6234179		
3	X: 102126 Y: 6233931		

Figure 12. Parking availability results - simple case. Green polygons: parkable space. Red polygons: non-parkable space due to house intersection, insufficient space or triangles.

5.4.2.2 Complex Case - Multiple Lines of Intersection

This section shows the results of properties that have multiple consecutive lines that intersect the property and the road. In the examples shown in Figure 13, the lines of intersection are treated individually using the method mentioned in Section 3.5.2. The program is capable of processing various situations, including when the house intersects multiple rectangles, as shown in Figure 13b ID 4. Figure 13 shows 4 different situations that can occur in the same property with a slight difference of house location.

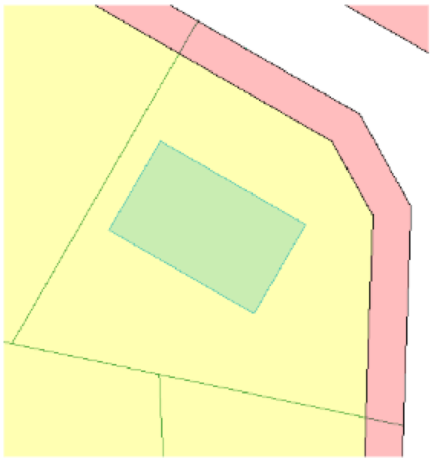
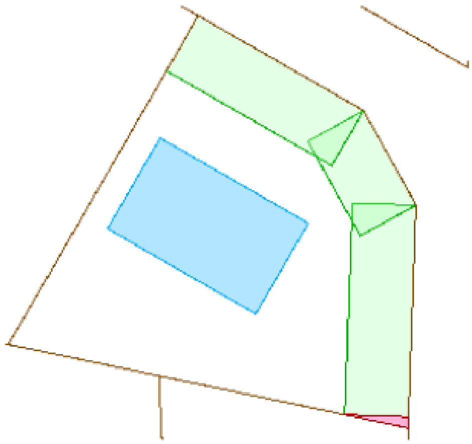
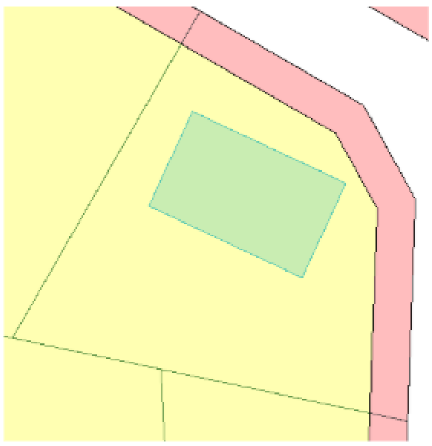
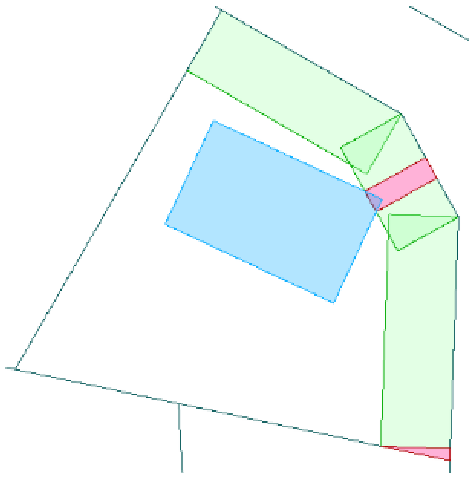
<table>
<tr><th>ID</th><th>Coordinates</th><th>House location</th><th>Parking Availability</th></tr>
<tr><td colspan="4">ID 1: The house does not intersect with any buffered rectangles and they all have width/length greater than 2.5m. The triangle is colored red because it is not suitable for parking.</td></tr>
<tr><td>1</td><td>X: 102120
Y: 6234135</td><td></td><td></td></tr>
<tr><td colspan="4">ID 2: There are 2 rejected polygons, one of them is a triangle, the other one intersects with the house.</td></tr>
<tr><td>2</td><td>X: 102123
Y: 6234137</td><td></td><td></td></tr>
</table>

Figure 13a. Parking availability results - complex case. Green polygons: parkable space. Red polygons: non-parkable space due to house intersection, insufficient space or triangles.

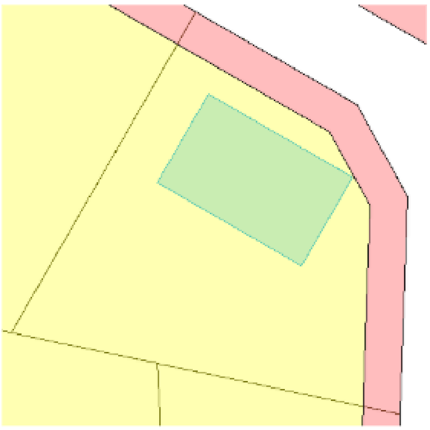
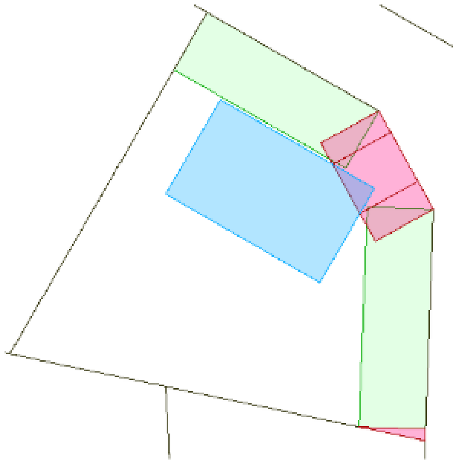
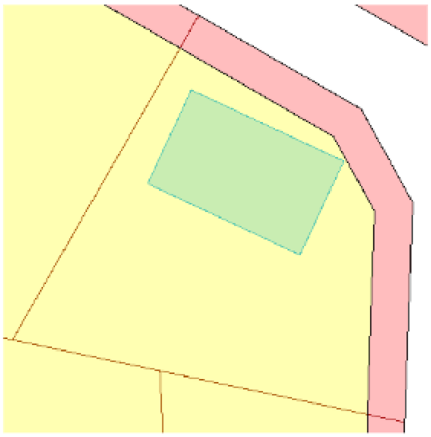
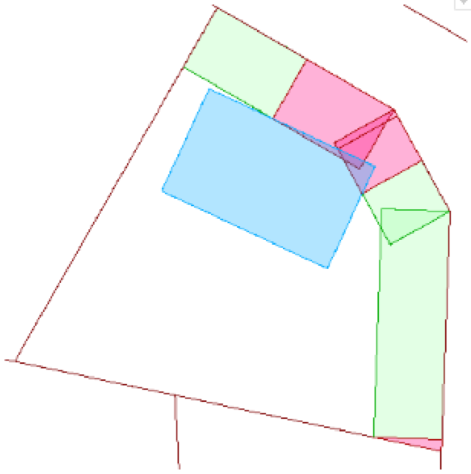
<table>
<tr><th>ID</th><th>Coordinates</th><th>House location</th><th>Parking Availability</th></tr>
<tr><td colspan="4">ID 3: There are 4 rejected polygons. One of them is a triangle, the other 3 either intersect the house or have width/length less than 2.5m.</td></tr>
<tr><td>3</td><td>X: 102124
Y: 6234138</td><td></td><td></td></tr>
<tr><td colspan="4">ID 4: There are 4 rejected polygons. One of them is a triangle, two of them intersect the house, the last one has a width/length less than 2.5m.</td></tr>
<tr><td>4</td><td>X: 102123
Y: 6234139</td><td></td><td></td></tr>
</table>

Figure 13b. Parking availability results - complex case. Green polygons: parkable space. Red polygons: non-parkable space due to house intersection, insufficient space or triangles.

5.5 Algorithm Overview

In order to check if the program can process all four building requirements at the same time, four tests with different coordinates are inputted into the algorithm. The algorithm performed compliance checking independent of the results of individual building requirement. Figure 14 indicates the automation results of building height, roof angle, building placement and parking availability.

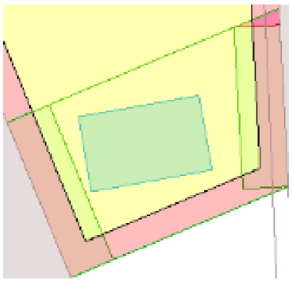
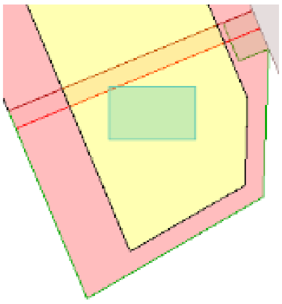
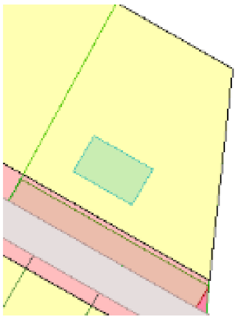
ID	Coordinates	Building Height	Roof Angle	Building Placement	Parking Availability	Program Outcome
1	X: 102133 Y: 6233898	✕	✓	✓	✓	
2	X: 102125 Y: 6233811	✕	✓	✕	✓	
3	X: 102057 Y: 6234068	✕	✓	✓	✓	
4	X: 102122 Y:6234176	✓	✓	✓	✓	

Figure 14. Algorithm results for building requirements: building height, roof angle, building placement and parking availability.

5.6 Evaluation of Results

Throughout the development of the algorithm, numerous tests have been made to validate the program's functionality. Of those tests, 20 are chosen to represent the results in this thesis report. All test results have been manually validated by using measurement tools in

FME. The evaluation proves that the algorithm can successfully detect and assess building height, roof angle, building placement and parking availability requirements.

6. Discussions

6.1 Findings

The results in this study demonstrate that it is possible to automate the checking of house placement. Nine tests with different parameters are applied to the algorithm and the results show an accuracy of 100%. The actual outcomes match the expected outcomes and the results are satisfactory. The developed algorithm is able to correctly identify whether or not a house's replacement is according to regulations, and is also able to represent the results both numerically and visually as shown in table 1, Figure 11, Figure 12 and Figure 13 and Figure 14.

The results in this study also demonstrate that it is possible to automate the checking of parking availability. Seven tests with different parameters are applied to the algorithm, some combinations are more simple and some are more complex. The results from the testing are accurate and conform to the expectations. Less obvious results, such as red rectangles in Figure 13b ID 3 and ID 4 are validated by using measurement tools in FME. The algorithm has proven to work effectively in identifying parkable and non-parkable space in a property.

In general, the aims of the study are fulfilled with satisfactory results. To my knowledge, this thesis is the first to explore house placement and parking availability compliance checking automation. The findings in this thesis fill existing gaps in the building permission field. The developed algorithm extends the possibility to automate building permission and provides new knowledge and contribution to the construction and planning industry.

6.2 Building Requirement Relationships

In this study, it is important to note that house placement and parking availability requirements are interconnected. In a common building permit compliance checking, the main limiting factors are the regulations and laws. Building requirements, such as height, footprint area and house placement are regulated and determined by the regulations. In this study, building requirements are limited by both regulations and other building requirements. For example, parking availability is regulated by the municipality and affected by the placement of the house. If the house is too close to the boundary, then parking space is limited and can result in insufficient parking space. Considering insufficient parking space is not ideal, house placement needs to be adjusted. Since building requirements are interconnected, if an owner wants to place a house close to the road, then parking space might be limited. Contrarily, if the property owner wants to have space for many cars, then house placement might not be too close to the road. While abiding to laws and regulations, a balance between these two requirements is recommended, but is dependent on personal preferences and priorities. In this study, it is important to be aware of the geospatial relationships between the house and the nearby built environment, as well as the relationships between each building requirement. Building requirements, such as house placement and parking availability in this case, have a more complex relationship than a simple check against the regulations. This is important to know for future building permit automation of other building requirements.

6.3 Implementation Difficulties

One of the biggest obstacles in this study is defining parkable and non-parkable space in a property. The difficulty lies in the fact that the property and the house can have different shapes and placement, as well as different proximities to the road. The proposed approach of using a buffer is a suitable approach because it can visually identify potentially parkable and non-parkable space. With the buffer, it is easier to detect and validate the requirements needed for parking.

Another obstacle is to ensure that all parkable space are rectangles because parking should be perpendicular to the road. Since the property can have different shapes, the buffer of potentially parkable space might be a tetragon or trapezoid. There are different ways to transform polygons to rectangles, but the calculation can be mathematically heavy. There is currently no transformer in FME that can handle this type of transformation due to its complexity. So it is difficult and time consuming to find an adequate solution. This study utilizes geometrical relationships to execute polygon transformation using a combination of different FME transformers. The algorithm achieves the desired results without complicated mathematical calculations.

6.4 Future Improvements

6.4.1 Maximum Number of Parking Spots

An interesting addition to this thesis is to add a section about finding the maximum number of parking spots per property, with regards to house location. This extra step is a great addition since all parkable space is already defined as rectangles. The calculation is simple if there is only one line of intersection between the house and the road. For example, if the width/length is 10m, then 10 / 2.5 = 4, the property can fit a maximum of 4 parking spots. The calculation gets complicated if there are multiple lines of intersection. If these lines are connected, then there are overlapping parking space. The overlapping area may or may not hinder the maximum number of parking spots, it depends on the length and angle of the intersected lines, which can be complex. Identifying the maximum number of parking spots is not within the scope of this thesis, but it is an interesting addition that is worth investigating. Since it adds useful information to the building permit, it is recommended as a future improvement.

6.4.2 Drag and Drop Option

In order to move a house to a desired location, users must manually input the coordinates of the house into the algorithm. Sometimes rotation of the house is also necessary for it to fit perfectly in a tight area. A drag and drop option will be a lot more convenient and intuitive to use, especially for users without any programming experience. However, it is not very clear whether or not FME supports a drag and drop function. Therefore, this feature is recommended as future improvement.

6.5 Online Application Service

This study advocates for an online application service that automate parts of the building permit checking process. The online service should check the building requirements mentioned in this study - house placement and parking availability. Other common building requirements such as building height and footprint area can also be included in the service. If all conditions are met, then the application proceeds for further processing. If conditions are not met, then the application is rejected and the applicants must make changes to their design. Rather than giving building specialists the responsibilities of checking basic building requirements, the service alleviates the burden and takes care of it for applicants. This redistributes the responsibilities of common rule checking to applicants and reduces the amount of work for building specialists, so they only need to manually check floor plans and site plans that pass basic building requirements. This potential online service can also become useful for building professionals to validate their plans instantly through a simple plan checking system. It minimizes the amount of building regulations that must be manually checked by building specialists and reduces the workload for the municipality. Since workload is reduced, the municipality can allocate the time and resources to other building related sectors. Additionally, process time will be faster since basic rule checking can be done online by applicants.

7. Conclusions

This thesis demonstrates the automation and rule checking of specific building requirements in Sweden. During the automation process, it is important to note that some building requirements are interconnected. This study takes into consideration both the geospatial relationships between the house and the surrounding areas, as well as the relationship between different building requirements. The developed algorithm successfully automates the compliance checking of building height, roof angle, house placement and parking availability to different levels. The aim of the thesis is achieved because the developed algorithm can successfully:

- ➢ Determine if house placement is according to standards.
- ➢ Check if house placement meets the minimum distance requirement and display results both numerically and visually.
- ➢ Identify parkable and non-parkable space in a property with regards to house replacement.

Results in this study are satisfactory, indicating that the algorithm can accurately and effectively detect whether or not the house features fulfill the regulations in the detailed development plans. In conclusion, this thesis proves that it is possible to automate the checking of house placement and parking availability in Sweden. Further research is highly recommended to automate less explored building requirements because automating construction permission is efficient and beneficial for societal and economic development. It is an important topic that will continue to gain more recognitions internationally.

References

Architectural Institute of Korea. 2016. Ebook. Seoul, Korea: Architectural Institute of Korea.

Arroyo Ohori, K., A. Diakité, T. Krijnen, H. Ledoux, and J. Stoter. 2018. Processing BIM and GIS Models in Practice: Experiences and Recommendations from a GeoBIM Project in The Netherlands. ISPRS International Journal of Geo-Information 7: 311. MDPI AG. doi:10.3390/ijgi7080311.

Building and Construction Authority. 2013. BCA wins Singapore Quality Award for business excellence.

Doing Business in a More Transparent World. 2012. Doing Business. The World Bank. Retrieved 26 June 2019, from https://www.doingbusiness.org/content/dam/doingBusiness/media/Annual-Reports/English/DB12-FullReport.pdf

Doing Business Training for Reform. 2019. Doing Business. The World Bank. Retrieved 26 June 2019, from https://www.doingbusiness.org/content/dam/doingBusiness/media/Annual-Reports/English/DB2019-report_web-version.pdf

Donkers, S., H. Ledoux, J. Zhao, and J. Stoter. 2015. Automatic conversion of IFC datasets to geometrically and semantically correct CityGML LOD3 buildings. Transactions in GIS 20: 547-569. Wiley. doi:10.1111/tgis.12162.

Ease of Doing Business Reforms 2014-18. 2018. Ebook. Ministry of Commerce and Industry: Department of Industrial Policy and Promotion.

Floros, G., I. Pispidikis, and E. Dimopoulou. 2017. Investing Integration Capabilities Between IFC and CityGML LOD3 for 3D City Modelling. ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences XLII-4/W7: 1-6. Copernicus GmbH. doi:10.5194/isprs-archives-xlii-4-w7-1-2017.

Guide Permits and Notifications. 2018. Boverket. Retrieved 26 June 2019, from https://www.boverket.se/en/start/building-in-sweden/swedish-market/laws-and-regulations/permits-notifications/guide-permits-notifications/

Kim, H., J. Lee, J. Shin, and J. Choi. 2019. Visual language approach to representing KBimCode-based Korea building code sentences for automated rule checking. Journal of Computational Design and Engineering 6: 143-148. doi:10.1016/j.jcde.2018.08.002.

Lee, H., J. Lee, S. Park, and I. Kim. 2016. Translating building legislation into a computer-executable format for evaluating building permit requirements. Automation in Construction 71: 49-61. Doi:10.1016/j.autcon.2016.04.008.

Macit İlal, S., and H. Günaydın. 2017. Computer representation of building codes for automated compliance checking. Automation in Construction 82: 43-58. Doi:10.1016/j.autcon.2017.06.018.

National Building Codes and standards of Korea. 2019. R-nbck.or.kr. Retrieved 15 May 2019, from http://r-nbck.or.kr/en/Sub/Sub_02_02.aspx.

Olsson, P., J. Axelsson, M. Hooper, and L. Harrie. 2018. Automation of Building Permission by Integration of BIM and Geospatial Data. ISPRS International Journal of Geo-Information 7: 307. MDPI AG. doi:10.3390/ijgi7080307.

Papadonikolaki, E., and H. Wamelink. 2017. Inter- and intra-organizational conditions for supply chain integration with BIM. Building Research & Information 45: 649-664. Informa UK Limited. doi:10.1080/09613218.2017.1301718.

Permits and Processes. 2019. Building Sweden. Retrieved 13 February 2019, from https://www.buildingsweden.com/permission-to-build

Planning and Building Act (2010:900). 2010. Sweden: Swedish National Board of Housing, Building and Planning.

Planning Process. 2018. Boverket. Retrieved 13 February 2019, from https://www.boverket.se/en/start/building-in-sweden/swedish-market/laws-and-regulations/planning-process/

State Council reforms approval system of construction projects. 2019. English.gov.cn. Retrieved 20 May 2019, from http://english.gov.cn/policies/latest_releases/2019/03/26/content_281476580818292.htm

Stouffs, R., Tauscher, H. and Biljecki, F. (2018). Achieving Complete and Near-Lossless Conversion from IFC to CityGML. ISPRS International Journal of Geo-Information, 7(9), p.355.

Tan, X., A. Hammad, and P. Fazio. 2010. Automated Code Compliance Checking for Building Envelope Design. Journal of Computing in Civil Engineering 24: 203-211. American Society of Civil Engineers (ASCE). doi:10.1061/(asce)0887-3801(2010)24:2(203).

Van Berlo, L., T. Dijkmans, and J. Stoter. 2013. Experiment for Integrating Dutch 3D Spatial Planning and BIM for Checking Building Permits. ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences II-2/W1: 279-284. Copernicus GmbH. doi:10.5194/isprsannals-ii-2-w1-279-2013.

Whyte, J., and T. Hartmann. 2017. How digitizing building information transforms the built environment. Building Research & Information 45: 591-595. Informa UK Limited. doi:10.1080/09613218.2017.1324726.

Zhu, J., Wang, P. and Wang, X. (2018). An Assessment of Paths for Transforming IFC to Shapefile for Integration of BIM and GIS. 26th International Conference on Geoinformatics.