

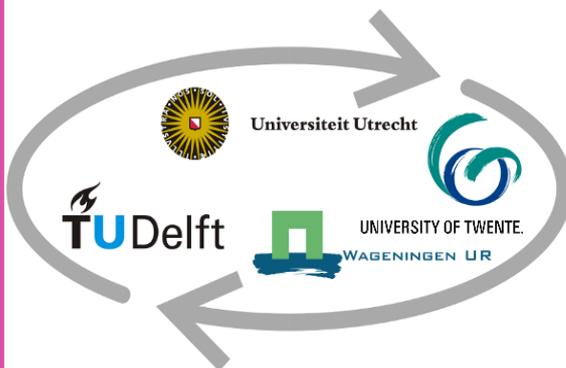
***Modelling Urban Transformations:  
A Spatio-Temporal Analysis of Building Changes in  
the Conservation Area of Utrecht***

***Final report***

Boris Leendertszoon Beije  
21-02-2025

Supervisors:

Prof. Dr. Ir. Peter van Oosterom – TU Delft  
Prof. Dr. Ana Pereira Roders – TU Delft



## Abstract

This thesis studies transformation of urban environments, taking the conservation area of Utrecht as a case study. It aims to explore methods for vectorization, object matching and change detection of building data across different time periods. By studying these three processes together, this study aims to develop a holistic approach for linking historical and modern building data, bridging the gap between past and present datasets and offering a pathway for more integrated analyses of the built environment. This research found that combining manual and automatic vectorization provides the most solid foundation for extracting buildings from historical cadastral maps. For object matching, a 1:1 match using building centroids, supplemented by 'contains' and 'within' spatial relationships, proved most effective in matching BGT Pand building versions. A rule-based change detection method, refined through 1:m spatial joins, was the most reliable for identifying building changes in the BGT Pand dataset between 2016 and 2024. It can be concluded that relatively many changes have occurred in the conservation area since the beginning of the BGT Pand version history in 2016, despite being an area that has strict instructions for the modification of buildings.

## Acknowledgements

Presented here is my GIMA MSc thesis, the culmination of a six-month-long research process. The past half a year has been an interesting process in which I have learned much about the research fields of map vectorization, object matching and change detection. A highlight was the OpenMapsMeeting in The Hague, which demonstrated that historical map data is anything but outdated. I would like to express my gratitude towards my supervisors Prof. Dr. Ir. Peter van Oosterom and Prof. Dr. Ana Pereira Roders for assisting me in the processes of researching this fascinating topic. To Wim Florijn and Kadaster for providing me their VeCTOR pipeline for the automatic vectorization process of the historical cadastral map. To Dr. Ir. Martijn Meijers, Dr. Nan Bai and Dr. Azarakhsh Rafiee of TU Delft for assisting me in the building vectorization process. Finally, I would like to thank my GIMA comrade Levi Hage for his mental support and in helping me painstakingly vectorize the buildings by hand. Enjoy reading!

Boris Leendertszoon Beije

# Table of contents

List of figures .....	5
Glossary .....	6
1. Introduction .....	7
1.1 Research objectives .....	8
1.2 Research scope .....	9
1.3 Reading guide .....	9
2. Literature review .....	11
2.1 Digitization of Historical maps .....	11
2.1.1 Manual vectorization .....	11
2.1.2 Automatic vectorization .....	12
2.1.3 Hybrid vectorization .....	15
2.2 Building change typologies .....	15
2.3 Discrepancies between datasets.....	17
2.4 Object matching.....	18
2.5 Change Detection.....	19
2.6 Conservation area .....	20
2.7 Monumental status.....	21
2.7.1 National monuments.....	21
2.7.2 Provincial monuments.....	21
2.7.3 Municipal monuments .....	21
3. Proposed solutions.....	22
3.1 Case study area .....	22
3.2 Data availability.....	23
3.3 Building vectorization .....	25
3.3.1 Georeferencing.....	25
3.3.2 Manual vectorization .....	26
3.3.3 Automatic vectorization .....	26
3.4 BGT Pand.....	28
3.4.1 Bi-Temporal model .....	28
3.4.2 Temporal component in the BGT Pand .....	28
3.5 Object matching.....	29
3.6 Change Detection.....	30
3.6.1 Area difference ratio .....	31
3.6.2 Refinement step .....	32
3.7 Accuracy assessment .....	32
3.8 Historical change detection .....	33
4. Results .....	34

4.1 Building vectorization .....	34
4.1.1 Georeferencing.....	34
4.1.2 Manual vectorization.....	35
4.1.3 Automatic vectorization .....	37
4.4 Object matching.....	40
4.5 Change detection .....	41
5. Discussion.....	48
5.1 Building vectorization .....	48
5.1.1 Georeferencing.....	48
5.1.2 Manual vectorization.....	48
5.1.3 Automatic vectorization .....	48
5.2 Object matching.....	49
5.3 Change detection .....	50
5.4 Limitations .....	51
6. Conclusion.....	53
7. Recommendations .....	54
Bibliography .....	55
Appendices.....	61
Appendix A: The rule system for change detection. Zhou et al., 2018. ....	61
Appendix B: Map of Utrecht in the twelfth century. Van Der Vlerk, 1983. ....	62
Appendix C: 1965 large scale cadastral map. The National Cadastre, 1965. ....	63
Appendix D: Field sketch and VeCTOR model output. Franken et al., 2021. ....	64
Appendix E: Code for object matching I .....	65
Appendix F: Code for object matching II.....	66
Appendix G: Code for change detection .....	67
Appendix H: Code refinement step I.....	69
Appendix I: Code refinement step II .....	70
Appendix J: Output refinement step.....	71
Appendix K: Demolished- and newly constructed buildings. Google, n.d. ....	72

## List of figures

Figure 1: Graphic illustration of the key steps in developing the proposed solutions .....	9
Figure 2: Relationship between AI, ML, DL, DNN and CNN. Zhang et al., 2022. ....	13
Figure 3: Example of a CNN for the vectorization of historical map data. Heitzler & Hurni, 2020.....	14
Figure 4: Identified change typologies in the BGT Pand between the earlier epoch (left) and the later epoch (right) .....	16
Figure 5: Discrepancies between OSM (green) and TOP10NL (cyan), with overlaps in brown. Zhou et al., 2018. ....	17
Figure 6: Controlled building alignment by centroid (1965 cadastral map in yellow, BGT in blue).....	19
Figure 7: Graphic Illustration of building vectorization from the 1965 cadastral map (above) and object matching with change detection in BGT Pand (below). ....	22
Figure 8: Research area .....	23
Figure 9: Global pipeline stages of the VeCTOR model. Franken et al., 2021.....	27
Figure 10: Decision tree for building change detection .....	31
Figure 11: GCPs in the research area (left), georeferenced historical map (right) .....	34
Figure 12: Overview of the manually vectorized buildings of the 1965 cadastral map .....	35
Figure 13: Examples of RGB images and corresponding binary masks .....	36
Figure 14: VeCTOR pipeline line segment output .....	37
Figure 15: VeCTOR outputs with decreasing confidence levels (0.5 top left, 0.05 top right, 0.005 bottom left, 0.0005 bottom right). ....	38
Figure 16: VeCTOR output difference when changing input image orientation.....	39
Figure 17: Object matching .....	40
Figure 18: The detected building version changes in the conservation area of Utrecht, between 2016 and 2024 .....	41
Figure 19: Demolished building.....	43
Figure 20: New buildings.....	44
Figure 21: Partly demolished building.....	44
Figure 22: Expanded/merged buildings .....	45
Figure 23: Timelapse of BGT building changes between 2016 and 2024.....	46
Figure 24: Demolition falsely identified as merge.....	52
Figure 25: Graphic illustration of recommended research directions.....	54

## Glossary

Term	Definition
ADR	Area Difference Ratio: A metric that resembles the relative change in area size between old- and new building versions.
Automatic vectorization	Methods that rely entirely on algorithms or machine learning models to convert raster data into vector data without human intervention during the extraction process (Chen et al., 2024).
BAG	Basisregistratie Adressen en Gebouwen: The Dutch national register for addresses and buildings, which contains information about the locations, addresses, and attributes of buildings (Kadaster, n.d.-b).
BGT	Basisregistratie Grootchalige Topografie: The Dutch national register for large-scale topography, containing detailed geographical information about buildings, roads, and other infrastructural elements (Kadaster, 2024).
Building change typology	A classification system that categorizes different types of changes in buildings over time.
Building versions	Different instances of a building in a dataset, reflecting administrative updates, measurement adjustments, or actual physical changes.
Centroid	The geometric centre of a polygon or feature, typically used to represent the "centre" of an object in spatial analyses.
Change classes	Categories used in change detection to classify different types of changes between two building versions.
Change detection	The process of identifying and analysing changes in spatial or temporal datasets (Matikainen et al., 2010).
Conservation area	Groups of immovable properties that are of general interest due to their beauty, their mutual spatial or structural cohesion, or their scientific or cultural-historical value, and in which one or more monuments are located" (Rijksdienst voor het Cultureel Erfgoed, n.d.).
Digitization	The process of converting physical data sources into digital formats.
False positives	Errors in data or analysis where a feature or change is mistakenly identified, even though it does not actually exist (Matikainen et al., 2010).
Field validation	The process of verifying data by collecting and comparing it with real-world observations or measurements.
Georeferencing	The process of precise alignment of the input spatial information through determining the position of input data in a spatial coordinate system other than its own (Cascón-Katchadourian & Alberich-Pascual, 2021).
GIS	Geographic Information System: A system for capturing, storing, analysing, and displaying spatial and geographic data, often used for mapping, analysis, and decision-making (Heywood et al., 1999).
Large-scale cadastral maps	Detailed maps that depict land parcels and property boundaries.
Manual vectorization	Methods that rely on manually tracing to convert raster data into vector data (Chen et al., 2024).
Object joins	Relational operations in spatial databases where attributes from one object (e.g., building) are matched to another object. A 1:1 join links one object to another, while a 1:m join links one object to multiple others (Zhou et al., 2018).
Object matching	the identification and matching of the same objects in different data sources (Zhou et al., 2018).
Pand	Smallest functionally and structurally independent unit that is directly and permanently connected to the earth and that can be entered and locked (Geonovum, 2022).
RMSE	Root Mean Square Error: A metric for assessing the accuracy and precision of the georeferencing process (Brovelli & Minghini, 2012).
Training data	A set of data used to train machine learning models or algorithms, which helps the model learn to classify or predict features in new data.
VeCTOR pipeline	A model developed by Kadaster for converting JPEG images of field sketches into digital vectorized networks of geometric observations (Franken et al., 2021).
Vectorization	the process of transforming scanned or rasterized graphical representations of geographic entities into a vector format which can be edited using GIS software, to be better indexed, georeferenced, and analysed spatially (Picuno et al., 2019).

# 1. Introduction

---

Understanding the transformation of urban environments is vital for a wide variety of research fields and disciplines, including urban planning, effective land use management and cultural heritage preservation (Drolias & Tziokas, 2020; Petitpierre & Guhenec, 2023; Liu et al., 2024). Information on the historical organization of urban landscapes is widely recorded in historical documents such as city archives and registers, notarial deeds and cadastral maps. Especially the latter provide valuable sources for the purpose of analysing spatial developments, as they are the most reliable documentations of cities on a fine scale level (Kruizinga & van Rosmalen, 1997). They therefore allow for the comparison of parcel boundaries and building outlines over time, enabling for studying cities' building structures across time periods. The use of geographical information systems (GIS) has increasingly been used to display, manage and most importantly analyse historical geographical data, including digitized cadastral maps (Femenia-Ribera et al., 2022). One of the biggest challenges in realising the full potential of historical cadastral maps lies in their vectorization, the process of converting scanned paper maps into vector data. Related topics like the ambivalence between using traditional manual vectorization (more accurate but not scalable) and modern automatic methods (less accurate but largely scalable) are recurring within research and organizations alike (Chen et al., 2024; Chen et al., 2020a). Building on data derived from the vectorization of historical maps, the research field of change detection allows for the analysis of changes on building level through time. Thereby, this process essentially enables the changes of buildings in urban areas to be tracked over time, offering detailed insights into the changing urban landscape in nature and pace. Hereby, these cadastral sources can be used to provide a nuanced understanding of historical contexts and narratives that are vital for cultural heritage preservation (Liu et al., 2024). Despite the importance and interconnectedness of both these processes, they are rarely researched in relation to each other.

The field of cultural heritage management and preservation is increasingly employing GIS as it integrates three essential components of cultural research: data acquisition, spatial analysis, and landscape reconstruction (Yao et al., 2023). Despite this increase, there remains significant untapped potential for the application of GIS within this field. Liu et al. (2024) for instance plead for increased integration and use of digital technologies, and specifically GIS, for the conservation of cultural heritage. They go on to discuss that, in addition to using GIS to study themes such as climate change, risk management and values, using it to study the historical development context of heritage holds unholstered potential. Their bibliometric analysis highlights the importance of strengthening efforts in the historical field of cultural heritage conservation, including the use of historical maps, overviews, and data. Such historical data is essential for evaluating the development of cultural heritage and its relation to broader spatial characteristics in cities. By placing cultural heritage within historical time and space, transformation processes and spatial characteristics are revealed, thereby also unveiling nuances regarding the zeitgeist of historical data. Similarly, Huang (2024) discusses the potential of using GIS in heritage studies for the digital recording and document management of heritage, as well as for interpreting historical data to deepen the understanding of heritage history and the cultural context.

The historical city centre of Utrecht, designated as conservation area (*beschermde stadsgezicht*) since 1975, is used as a case study to explore the application of the researched methods and to study how buildings change over time. In the past, the structure of historical cities in the Netherlands underwent profound transformations, driven by shifts in societal needs, technological advancements, and evolving urban functions (Dolfin et al., 1989). Existing buildings that no longer met the whims of social and urban needs were regularly redeveloped or demolished without consideration of their artistic value or significance for the streetscape and broader urban structure (Hulsman, 2020). In their stead, new

structures were constructed that accommodated new uses to better align cities with the evolving demands of their populations. This only changed with the introduction of the first national Heritage Law in the Netherlands in 1961, following a temporary law from 1950 (Tillema, 1975). This law introduced a careful legislative procedure that made it impossible for municipal authorities to easily grant demolition permits for listed heritage (hereinafter built heritage). The transformation of cities on a building level can be seen as a clear indicator of these functional shifts as cities adapt to societal needs. While this case study shows the potential for change analysis in heritage conservation, the methods studied in this thesis can be applied to urban transformations in other contexts.

## 1.1 Research objectives

This thesis aims to model urban transformation, taking the city of Utrecht as a case study. Through the perspective of building changes within the conservation area of the medieval city centre it seeks to study the preservation and transformation of the urban landscape on the building level. Its aim is twofold. Firstly, it focuses on different methods for the vectorization of historical cadastral data at the building level, with the goal of exploring how historical maps can be used for modern scientific analyses. Secondly it studies how to most effectively compare cadastral data from different temporal versions of the BGT Pand dataset (*Basisregistratie Grootchalige Topografie*), aiming to understand how building changes can be classified over time. The main research question that this thesis will answer is:

*"How can the processes of vectorizing historical cadastral maps and detecting building-level changes over time be combined to analyse the urban transformations of the historical city centre of Utrecht?"*

To further guide the research process, three sub questions have been formulated. The first question is aimed at identifying the methods most appropriate for the vectorization of large-scale cadastral maps. The vectorization process of historical cadastral data is necessary for turning it into usable vector data, enabling further spatial analysis at the building level. By studying advantages and limitations associated with techniques for vectorizing historical map data, the most suitable methods are determined. Hereby, this thesis seeks to advance the use of historical cadastral maps for modern applications.

### 1. *"What methods are most effective for the vectorization of historical cadastral data?"*

The second sub-question addresses the challenge of object matching, linking corresponding buildings from different building versions. Considering that building changes occur over time, this step binds different time steps together, enabling versions of buildings to be correctly matched with one another. Thereby it is crucial for effectively performing the subsequent change detection. This question explores methods for matching building footprints, considering geometric, topological, and attribute-based factors to ensure reliable comparisons. The constructed method is finally used to match building versions in the BGT Pand dataset.

### 2. *"What are the most effective approaches for matching different versions of buildings?"*

The third and final sub-question studies methods to detect changes between different versions of buildings in a temporal context. It examines how buildings have changed over time by identifying the following transformations: demolition, construction, partial demolition, expansion, merging, and splitting. Different approaches exist for classifying changes in buildings over time. Through studying multiple forms of change detection, this research aims to find the most effective methods for comparing building data from different versions. Finally, the BGT Pand dataset, is used as a proof of concept for the constructed change detection method over the past 8 years (2016 - 2024).

3. “What methods are most suitable for researching the historical destruction, construction and changes of buildings?”

## 1.2 Research scope

Considering that this thesis touches upon a myriad of interconnected research areas, various methods also exist for vectorization, object matching, and change detection. Although this thesis explores multiple approaches through an extensive literature review, the proposed solution only employs a selection of methods and data sources to ensure feasibility given the time constraint. To help clarify this focus, the following aspects are considered in scope and out of scope:

### In Scope:

- The georeferencing and partial manual vectorization of a historical cadastral map.
- The testing of an untrained machine-learning pipeline for automatic map vectorization.
- The development and evaluation of an object matching and change detection approach.
- The application of both approaches to analyse building changes in the BGT Pand dataset from 2016 to 2024.

### Out of Scope:

- The vectorization of historical maps beyond the test area selected for this research.
- The training of a machine-learning approach for vectorizing cadastral maps
- The application of the proposed object matching and change detection approaches beyond the BGT Pand dataset
- A detailed historical analysis of urban development patterns

Figure 1 shows the methodology used to construct the proposed solution. The process starts with reviewing the literature, followed by data collection. Then, the solution is designed and tested, after which iterative feedback loop to refine it until it meets the required standards. After this process, the solution is ready for implementation in the BGT Pand dataset.

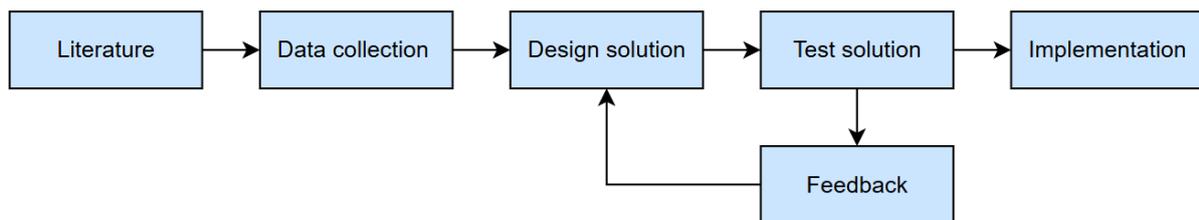


Figure 1: Graphic illustration of the key steps in developing the proposed solutions

## 1.3 Reading guide

In Chapter 2, the state-of-the-art is discussed. This includes sections on map vectorization, building change typologies and discrepancies between datasets. Then change detection is elaborated on, followed by a section on the conservation areas and monumental statuses in the Netherlands. In Chapter 3 the methods to address the research questions are discussed. First, the research area is described. Next, the chapter explores the vectorization of historical cadastral data, covering georeferencing, manual and automatic vectorization, the BGT data model, object matching, and change detection. Finally, the verification method of the results is explained, followed by a brief overview of what a change detection pipeline for comparing historical cadastral data to a new data source might look like. In Chapter 4, the results of the proposed solutions are presented. This is subdivided between

the vectorization of the historical cadastral map, including the georeferencing, manual- and automatic vectorization processes. Then, the outcomes of the object matching and change detection are illustrated and briefly explained. In Chapter 5, the outcomes from the results sections are interpreted and elaborated on more extensively within the context of the research. Some limitations of the used methodology are also included here. In Chapter 6, the objectives of the thesis are briefly restated, after which the identified sub questions and the main research question are answered. Finally, in Chapter 7, recommendations for future research based on this study are given.

## 2. Literature review

---

This theoretical framework will elaborate on methods used to vectorize maps in the related literature. Hereafter, it focusses on building change typologies and on how to deal with possible discrepancies between datasets. Then, the buildings change detection is discussed, followed by a concluding section discussing the concept of conservation areas and the various monumental statuses in the Netherlands.

### 2.1 Digitization of Historical maps

Efforts to make historical maps available online through digitization have been ongoing over the past decades fuelled by the realization that these documents should be available to the public (Levi, 2009). Additionally, they contain a wealth of information about topics that are still very much relevant, such as heritage preservation, spatial developments and the evolution of territory (Petitpierre & Guhenec, 2023). To extract geospatial information from these documents, the first step is scanning them; thereby making them available digitally (Drolias & Nikolaos, 2020). This work has resulted in the creation of numerous online databases containing historical maps that can be viewed, downloaded and are even georeferenced and overlaid onto digital maps (see for instance the OldMapsOnline environment of Utrecht University, (n.d.)). By scanning historical maps and making them digitally available, a raster image is created where every cell in the raster matrix is associated with a data value in the red, green and blue (RGB) bands, or binary in black and white. These values can then be used to extract information from the digital map, like spatial features such as building footprints, roads, and water bodies based on differences in pixel values. Raster data is particularly useful for analysing large scale spatial layout, measure distances and study land use changes over time. Additionally, raster data allows for the georeferencing of historical maps, enabling their overlay with modern digital datasets to study temporal changes.

Raster data also has distinct limitations that constrain its use in certain types of research. As the well-known saying within the GIS discipline, “raster is faster, but vector is corrector” suggests, raster data is particularly well-suited for examining large-scale spatial phenomena. However, it lacks the precision needed for detailed analyses on the object level. In these cases, vector data is more useful as it represents features as discrete geometric shapes (points, lines, and polygons). Additionally, associated attribute tables to the data allow for precise spatial queries and analyses. By converting raster images into vector data, researchers can label individual buildings, parcels, or infrastructure more accurately. This makes it easier to quantify changes and integrate historical data with modern GIS datasets (Drolias & Nikolaos, 2020). This is crucial for studies into transformations at a building level, as accurate boundary definitions are essential for researching trends in this field. To extract vector data from a raster image, map vectorization must be performed. Picuno et al., (2019) define map vectorization as the process of transforming scanned or rasterized graphical representations of geographic entities into a vector format which can be edited using GIS software, to be better indexed, georeferenced, and analysed spatially. Generally, the literature divides between three types of methods for map vectorization: manual, automatic and hybrid methods (Chen et al., 2024). These types will be further elaborated on in the following sub sections.

#### 2.1.1 Manual vectorization

Traditionally, the vectorization of historical map data is done manually, and it is still the most popular method when the map is small in coverage and time period (Chen et al, 2024). The process of manually vectorizing maps involves drawing the objects in a map digitally via a GIS or AutoCAD software. When a larger map, or a collection of maps must be vectorized, collaborative approaches like crowd sourcing are sometimes used, where a group of contributors help to speed up the process (Southall et al., 2017).

Despite that manual methods are still among the most used methods of map vectorization; it is associated with some considerable disadvantages. The largest drawback is that it is a very time-consuming process and therefore costly to implement on a large scale (Chen et al., 2024). Additionally, the quality of the results is largely dependent on the contributors' abilities to correctly draw the features in the maps. Furthermore, the lack of universally accepted methods and standards for manual vectorization make it difficult to compare the errors and results of projects using it as a method (Skaloš et al., 2011). To overcome these disadvantages and following advancements within fields as computer vision and machine learning, automatic and hybrid methods for the vectorization of historical maps has increasingly become an area of interest for researchers. The main difference between automatic and hybrid methods is that the former rely entirely on algorithms or machine learning models to convert raster data into vector data without human intervention during the extraction process. The latter, on the other hand, combines automated techniques with manual or supervised rule-based enhancements to correct errors or improve the accuracy of the model (Chen et al., 2024).

## 2.1.2 Automatic vectorization

### Colour and texture

Fully automatic methods for the vectorization of historical maps have been proposed by numerous researchers. Early methods of automatic vectorization mainly used RGB pixel values from the original raster images to classify objects in historical maps. This is done by using e.g. thresholding or region growing algorithms to separate different layers based on their associated colours, like blue for water bodies, green for vegetation and red for man-made structures (Dhar & Chanda, 2006; Petitpierre, 2020). Khotanzad and Zink (2003), for instance, make use of a method based on RGB values to extract and recognize geographical features from coloured topographic maps. Despite being relatively easy to implement, methods depending on raster RGB values for the vectorization of maps often work with a limited number of historical maps, like those with distinct colouring. Additionally, they often focus on few features, such as roads or parcels, while leaving out much of other map information. These approaches are thus difficult to adapt to many historical maps, especially those with complex shapes and layouts, such as urban areas (Chen et al., 2024). Furthermore, maps produced before the mid 19<sup>th</sup> century were generally printed in black only and those that were coloured have often faded due to the circumstances under which they were preserved. To overcome these limitations, some researchers have tried using textures, mainly focussing on features like hatched areas in historical maps. However, as methods targeting textures largely depend on size and rotation, they require customized parameterization per use case which has severe implications for their scalability and applicability (Petitpierre, 2020).

### Mathematical morphology

Other researchers like Chen et al., (2021a) use the morphological features of maps, such as lines, edges and closed polygons, to vectorize objects from historical maps. These methods rely on the principles of mathematical morphology (MM) to identify geometric features based on their shapes. In MM, operations such as dilation, erosion, opening and closing are used to analyse the structure of an input image; in this case a historical map (ESRI, n.d.-a). Techniques including MM are an often-employed method to account for incomplete edges, closing contours and recover other incomplete elements in maps (Petitpierre, 2020). A limitation of MM for vectorization is that it is highly susceptible to information overlays over the map objects such as map grids, background textures, text and symbol, which are very common in cartography.

## Neural networks

Neural networks, also referred to as classical- or shallow neural networks (SNN), are among the early method that have been widely employed and adapted to vectorize digitized raster maps (Petitpierre & Guhenec, 2023). SNNs are models made up of one or two layers of connected units called "neurons", which process information. The network takes input data, in this instance a historical map, and passes it through these layers to make a decision or prediction; to identify objects in input the map. Each neuron performs simple calculations, and the network learns by adjusting these calculations based on the output, so-called "backpropagation". This learning process makes the method a form of machine learning. The first instance of this method being used for the vectorization of historic maps was by Chen et al. in 1996. They used a SNN for the automatic extraction of parcels, text and rotated characters from scanned images of Chinese cadastral maps.

## Deep neural networks

Over the course of time, SNN have been refined and with the introduction of deep learning in recent years, they have evolved into deep neural networks (Ignjatić et al., 2018). These deep neural networks (DNN) outperform the SNN based on machine learning, considerably improving visual object and pattern recognition. The main difference between the two is that DNN are capable of learning higher-level, more complex and abstract features than their non-deep counterparts. This is achieved by combining feature learning and model building into one process. In a DNN, the model is built by selecting kernels and adjusting the model's parameters automatically through end-to-end optimization without human interference (Sze et al., 2017; Ignjatić et al., 2018). DNN are made up of many layers working in a non-linear way to transform the data into more abstract features as it moves further through the network, also referred to as "feedforward". The deeper in the network the input data is, the more complex patterns it can learn.

## Convolutional neural networks

Recent studies like that of Petitpierre et al. (2021) refer to the use of convolutional neural networks (CNN) for historical map vectorization. CNN are special forms of DNN that are the most widely used neural networks for computer vision tasks, including object detection (Ignjatić et al., 2018). The position of CNN within broader artificial intelligence (AI) is illustrated below in Figure 2.

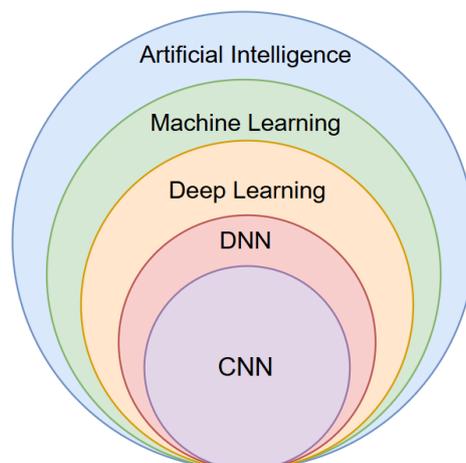


Figure 2: Relationship between AI, ML, DL, DNN and CNN. Zhang et al., 2022.

As in a DNN, data fed to a CNN passes through a number of hidden layers which learns the model to recognize features in the data. However, a CNN has fewer parameters and connections between its layers to better allow for two-dimensional image processing. It has three different types of layers through which input data, in this instance historical maps, passes: convolutional layers, pooling layers and fully connected layers (Vloulodimos et al., 2018). In the convolutional layers, kernels are used to extract local features by convolving input data, after which the pooling layers summarize and reduce the dimensionality of feature maps. Finally, in the fully connected layers perform high-level reasoning by combining outputs of the top convolutional layer, thereby converting the 2D feature maps into 1D features. This process for the vectorization of a historical map is visualised in Figure 3. To train a CNN for map vectorization, a dataset is often used that consists of input images of cut-out sections of the input map and corresponding binary images containing the buildings with pixel information of one, and zero otherwise (Heitzler & Hurni, 2020).

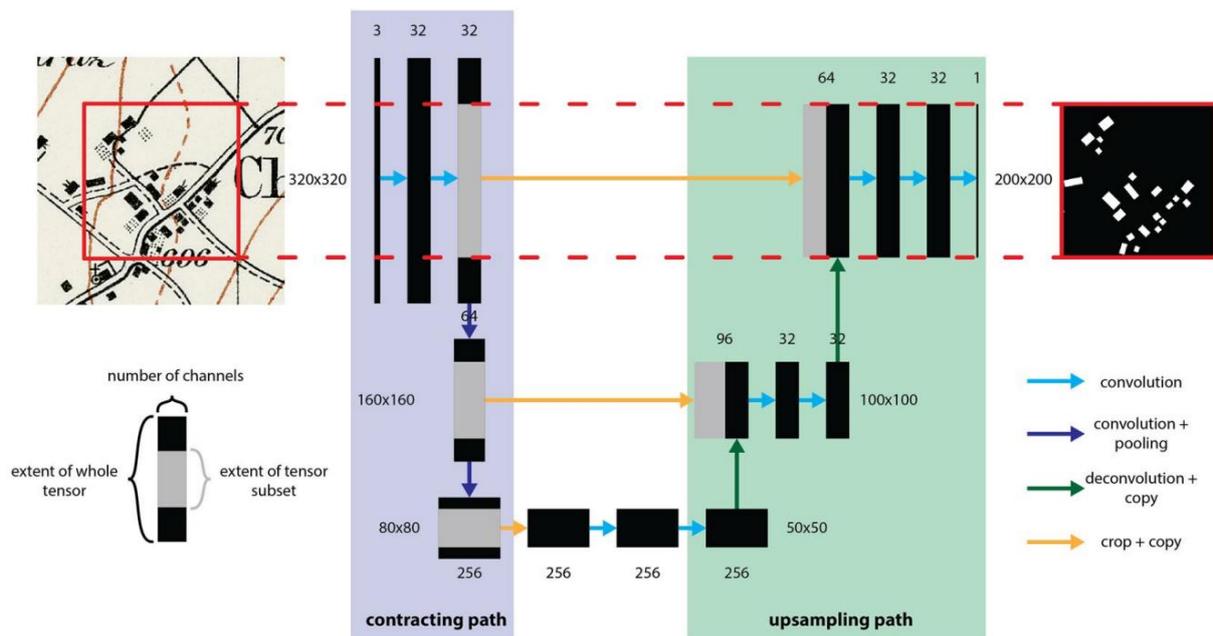


Figure 3: Example of a CNN for the vectorization of historical map data. Heitzler & Hurni, 2020.

Where more traditional computer vision algorithms have been used successfully for extracting information from very homogeneous maps, it is argued that CNN allows for the vectorization of more heterogeneous bodies of maps. Therefore, this method is very suitable when a more flexible approach is required, as is the case for studying multiple diverse historical maps in characteristics such as scale, hue, writing, scanning quality et cetera. However, CNNs are also characterized by some distinct weaknesses, such as the inability to detect shapes if only the outlines are illustrated in the input maps. To counter this, Chen et al. (2021b) combined the use of CNN with MM for segmenting the contents of historical maps. By doing so, the authors combined the strengths of CNN in efficient edge detection and filtering, with that of MM in the extraction of closed shapes. Additionally, Guidotti et al. (2018) highlight the challenges posed by AI-based models, including CNNs, which operate as black boxes, hiding their internal logic to the user. This opacity presents both practical and ethical concerns, as it prevents the formation of a standardized definition for the models' interpretation.

### 2.1.3 Hybrid vectorization

Hybrid vectorization combines automatic and manual methods, thereby addressing the limitations of both individual methods (Chen et al., 2024). Hybrid methods theoretically offer a suitable basis for digitizing historical maps, as they balance manual precision with automated scalability. An example of a hybrid method is derived from research by Budig et al. (2016) into a crowdsourcing project for the extraction of building footprints from historical maps of New York City. The three-year crowdsourcing project showed participants building footprint polygons which were automatically extracted from the input maps but often had errors. The participants were then tasked with deciding whether those building footprints were correctly extracted by the model, using the original subsection of the map as reference. If the outline was voted as incorrect more than three times, the outline was redrawn by a participant in a specially created tool in a subsequent step of the project. Each correction was made by multiple users, which were then aggregated by an algorithm into a consensus polygon for the corrected footprint. The resulting "consensus polygon" was validated against manually vectorized ground truth data and achieved a higher accuracy (96%) than the individual user polygons (85%). Despite leveraging strengths of both manual and automatic methods, hybrid approaches also have limitations (Chen et al., 2021a). Among the biggest drawbacks is that hybrid methods may prove to accomplish the opposite; being resource intensive while not generalizable for other datasets. This can be argued because crowdsourcing depends on factors such as a broad participant base, the participants' accuracy in vectorizing building outlines, creating and testing an environment for the crowdsourcing project et cetera. Additionally, used algorithms like that of Budig et al. (2018) are still very dependent on data being fed and additionally require parameter adjustments when used for vectorizing different maps, as does a CNN for instance. As a result, hybrid approaches are not yet common in the research area of historic map vectorization.

### 2.2 Building change typologies

When studying the changes to individual buildings on cadastral maps, it is essential to consider the potential changes that can occur. Such a topology for building changes encompasses all changes that can occur over time on a building level. Jovanović et al. (2022) make use of a simplistic typology containing three classes of building changes in their study comparing cadastral data to satellite imagery. These classes include new buildings that do not exist in the cadastre but are visible in the satellite data, demolished buildings that are still registered in the cadastre but demolished in the field and modified buildings where the base dimensions have changed in relation to the original records. Matikainen et al. (2010) propose a typology including five different classes in their article researching the automatic detection of changes in buildings for updating maps. These classes are defined to categorize the building status based on comparisons between the baseline building map and building outlines detected in subsequent time steps. The classes between which differentiation is made are:

1. **Unchanged buildings:** One building on the map corresponds to one in the building detection (1-1).
2. **Changed building:** A building exists in both datasets but has undergone modifications e.g., changes in size, shape, or orientation (1:1).
3. **New buildings:** No buildings on the map, one in the building detection (0-1).
4. **Demolished buildings:** A building is present in the existing map but is absent in the new building detection results (1:0).

5. **Complex changes:** One building on the map, more than one in the building detection (1-n), or vice versa (n-1). This can be a real change, or it can be related to generalization or inaccuracy of the map or problems in building detection (1-n/n-1).

However, in this typology, no differentiation is made between merging and splitting of buildings as both are grouped under the ‘complex changes’ class. Similarly, Hajiheidari et al. (2023) discusses five enrichment types for updating data in urban cadastral maps which are strongly related to those previously discussed. The enrichment types include adding (new parcel), deleting (demolished parcel), merging (two or more parcels joining), splitting (a parcel disintegrating into two or more), and changing (a parcel’s topology changing). Figure 4 shows the used building change typologies in green, derived from the literature. These are displayed over a base map which is used as reference for the location of the buildings in question.



Figure 4: Identified change typologies in the BGT Pand between the earlier epoch (left) and the later epoch (right)

### 2.3 Discrepancies between datasets

When conducting research into changes between datasets, one should consider possible discrepancies between the input data sources. A distinction can be made between absolute- and relative errors (Berk & Ferlan, 2016). Absolute errors refer to the direct difference between the mapped object and the real-life object and are typically measured in the actual units. Conversely, relative errors refer to the difference between the mapped object and the real-life object and are expressed as a proportion of the real-life size, often a percentage. This is especially relevant when comparing historical map data as these were manually drawn and surveyed. Traditional techniques for mapping such as triangulation relied on manually measuring distances and angles. In the production of early cadastral maps of the Netherlands, measuring chains of 20 to 30 meters in length, marked in 20-centimetre links were used by surveyors (Kruizinga & Van Rosmalen, 1997). Main measurement lines were based on triangulation and secondary lines were used to follow parcel boundaries. Measurements focused only on boundary mapping, with minimal points marked for straight borders and no systematic accuracy checks. For curved boundaries, only key points were measured, and intermediate lines were estimated by eye and memory. Church towers were used as reference points from which the maps were produced across large land areas. Consequently, the cadastral maps accumulated errors due to limitations in measurement precision, physical obstacles, and human error. This often led to inaccuracies in defining boundaries, especially further away from urban areas. As a result, absolute measurement errors of up to 40 meters are not uncommon in early Dutch cadastral maps according to Kruizinga and van Rosmalen (1997). In more recently produced digital maps inaccuracies between datasets also exist, albeit in much smaller scale. Zhou et al. (2018) study differences between heterogeneous data sources maintained by OpenStreetMap (OSM) and the Dutch Cadastre. Data managed by commercial data providers is often updated on a quarterly cycle, potentially leading to a situation where professional data may be outdated compared with their crowd-sourced counterparts. This is most apparent in locations where changes follow each other at a rapid pace, as is the case in urban areas (Fan et al., 2014). This is also apparent in OSM and the TOP10NL dataset managed by Kadaster, where small deviations exist between vector objects like buildings, as is illustrated in Figure 5.

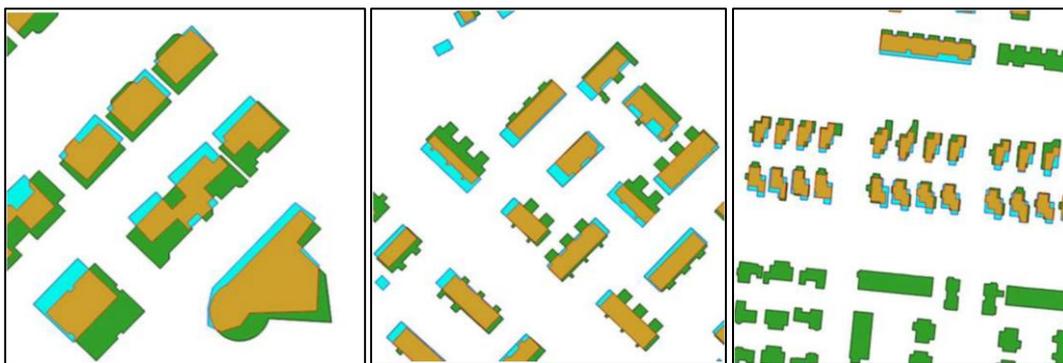


Figure 5: Discrepancies between OSM (green) and TOP10NL (cyan), with overlaps in brown. Zhou et al., 2018.

Another issue raised by Zhou et al. (2018) is that the discussed discrepancies between data sources are hard to distinguish from real-world changes. When attempting to overlay datasets, differences in level of detail (LOD) lead to fragmented polygons (as is seen in Figure 5), where areas of overlap, differences, and intersecting boundaries do not always represent physical changes. Therefore, a mere size-based analysis of polygons can be misleading: a fragmented polygon from OSM, for instance, could appear larger than in TOP10NL and suggest a modified building when in fact no change has occurred. Because

offsets between the datasets are often non-uniform, they cannot be fixed solely relying on geo-referencing adjustments. The random nature of offsets introduces a significant challenge with regard to the change detection of polygons. The challenges posed in the article become even more relevant when comparing digitized historical maps with modern building data. This can be argued because historical building data is often associated with simplified representations of buildings, varying boundary definitions, inaccurate measurements, and inaccuracies caused by digitization processes such as geo-referencing and rubber sheeting (Drolias & Tziokas, 2020; Kruizinga & van Rosmalen, 1997).

## 2.4 Object matching

As the name indicates, object matching involves the identification and matching of the same objects in different data sources. When spatial offsets between objects in the datasets are small and the data is of similar scales, a simple overlay analysis can be used to match objects (Matikainen et al. 2010; Zhou et al., 2018). However, when working with data that does not precisely overlap, this method is inadequate for object matching. Hajiheidari et al. (2023) employ a different technique for matching two datasets of the city of Tehran, Iran. As they use maps produced by different mapping agencies (the Municipality of Tehran and the Cadastre office of Iran Deeds and Property Registration Organization), the first process includes the preprocessing of the input data. This includes correcting topological errors and aligning the data scales. Hereafter, the polygons derived from the buildings in the maps are transformed into singular centre points. These points are used to generate buffers, which are employed to identify corresponding buildings between the two datasets. If the centre point of a Tehran Municipality Dataset parcel falls within the buffer zone of an Iranian Cadastral Organization Dataset parcel, the two are considered a match. This method allows for more flexibility to match buildings when they have large offsets but might lack in accuracy as opposed to an overlap analysis. Additionally, it does not account for the shape and size of the polygons. As a result, it might match parcels because the generated centre points are in close proximity, but in reality, have entirely different shapes or sizes. Therefore, this method might be combined with other matching techniques, manual inspection or additional features to account for ambiguous matches.

A technique to counteract the problem of boundary mismatches between datasets is to align buildings based on their centre of gravity (Zhou et al., 2018). This method is used to reduce discrepancies caused by deviations in the same objects in different datasets. It improves the reliability of shape comparisons in the consequent step of change detection by ensuring features align as best as possible, reducing offsets between buildings. An example of controlled alignment is demonstrated in Figure 6. It should be noted that centroid alignment is only applicable to single buildings or blocks of buildings with simple shapes (e.g. squares or rectangles). This limitation is caused by topological errors that might arise when aligning polygons by their centroids independently when the relative position of the buildings within the block are changed. Therefore, a block-level approach, like that used by Zhou et al. (2018), is essential for consistency when using this method.

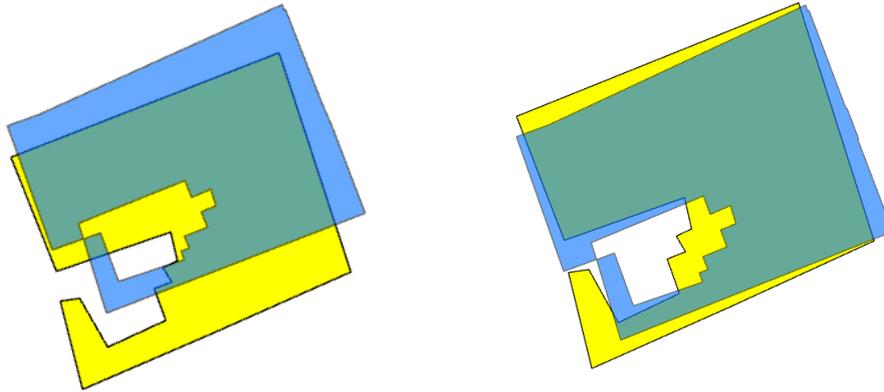


Figure 6: Controlled building alignment by centroid (1965 cadastral map in yellow, BGT in blue).

## 2.5 Change Detection

When two or more datasets of the same set of buildings are available, it is possible to perform change detection between the old and the new data. Multiple methods exist for the detection of changes between datasets depending on the type of data that is to be compared. When performing change detection between two- or multiple cadastral maps, objects vectorized from map data or remotely sensed imagery, techniques like overlap analysis and buffering can be used. Matikainen et al. (2010) use this technique by creating buffers around objects through morphological operations. The change detection rules allow small differences in the location and appearance of the same buildings over different datasets. Larger differences, however, are considered “true” changes or errors that require further examination. A rule-based method for change detection has some distinct advantages. Firstly, it facilitates accurate comparisons and reliable change detection across different temporal versions of a dataset. This is indicated by the findings of Matikainen et al. (2010), who found accuracies of 69%, 88% and 81% for new buildings, changed buildings and complex cases respectively. Additionally, a rule-based approach allows for flexible parameter finetuning based on the accuracy of the outputted results, as shown by Zhou et al. (2018). It should be noted that this method using object matching and overlap analysis can only be employed when buildings are accurately positioned.

Change detection performed by Hajiheidari et al. (2023) involves comparing the geometric properties (the area and number of nodes) of the matched polygons. If significant discrepancies are found between the matched polygons, the buildings are flagged as suspicious. This indicates that potential changes have occurred, or that inaccuracies are present in one of the datasets. This is automated using a logistic regression model for the flagging of suspicious buildings.

Zhou et al. (2018) present a rule-based change detection approach for analysing building footprints using multiple factors and methods. Before using these change rules, the authors converted the one-to-many (1:m) relations (multiple historical buildings corresponding to one modern building) to one-to-one (1:1) relations as this makes it easier to analyse the changes. This is done by aggregating the objects using the pair-wise aggregation tool in ArcGIS, which is stopped when too much empty space is enclosed, indicating physical change. Also, small buildings are aligned based on their centre points (as discussed in the previous paragraph), while bigger, more complex buildings are measured and aligned using a turning function. Hereafter, the morphology of the difference parts of the overlapping building footprints is analysed. If the difference is fragmented or has a thin shape, it may indicate an offset. For large buildings, the changes are analysed based on the absolute size of the difference. For

small buildings, both absolute and relative sizes are considered, because when a large percentage of a small building is flagged as changed, it might only be an offset between the datasets.

For potential changes, a constrained Delaunay triangulation is applied to generate a skeleton (centre lines) of the object, which are used to calculate the average length and width of the difference. A directed acyclic graph (DAG) structure is used to organize the change rules, where size and shape analysis are prioritized based on building size. Small buildings are more sensitive to boundary mismatches and are handled with a simpler rule set. The established rules of the model dictate that differences that are wider than a threshold of 5 meters are considered significant changes. If the changes are not wide enough, but large enough (larger than at least 100 m<sup>2</sup>, or 20% of the building) and in a compact form, this is also considered a change. Finally, the model accounts for the patterns and contextual information of buildings, as building configured in certain alignments (e.g. grid-like or linear patterns) are less likely to be modified than individual buildings. An alignment is identified as a homogeneous group of buildings that are evenly spaced and have similar forms, sizes, and regular layout (Zhang et al., 2013). This alignment recognition is done using Delaunay triangulation with a heuristic being used to adjust detected changes based on the consistency of building alignments between datasets, based on Zhang et al. (2011). If the alignment is maintained in both datasets, the buildings in the group were considered unchanged with a higher probability. The entire schematic model of the change detection model is illustrated in Appendix A. Three different variations of the model were used:

4. **Basic:** Basic geometric analysis without controlled alignment and morphology analysis
5. **Advanced:** Basic geometries with controlled alignment and morphology analysis
6. **Advanced + pattern recognition:** Basic geometries with controlled alignment and morphology analysis corrected by the pattern constraint.

This yielded the results illustrated in Table 1, where all values range from 0 to 1 with a higher value indicating a higher level of satisfaction. The performance was assessed using true positives (*TP*), false positives (*FP*), true negatives (*TN*), and false negatives (*FN*). Metrics include precision ( $TP / (TP + FP)$ ), recall ( $TP / (TP + FN)$ ), accuracy ( $(TP + TN) / (TP + FP + TN + FN)$ ), and Cohen's kappa (*k*).

Table 1: Performance of the model. Zhou et al., 2018.

Method	Precision	Recall	Accuracy	K
Basic	0.55	0.76	0.77	0.47
Advanced	0.82	0.87	0.90	0.77
Advanced + Pattern	0.87	0.87	0.92	0.81

It can be concluded that the model benefitted especially from including the controlled alignment and morphology analysis in the advanced method, while its performance only increased marginally when accounting for the pattern constraint.

## 2.6 Conservation area

In the Dutch national Heritage Law of 1988, a conservation area (*beschermd stads- of dorpsgezicht*) is described as: "Groups of immovable properties that are of general interest due to their beauty, their mutual spatial or structural cohesion, or their scientific or cultural-historical value, and in which one or more monuments are located" (Rijksdienst voor het Cultureel Erfgoed, n.d.). A conservation area can thus be described as an area within a city with a special cultural-historical character. The protection of these areas aims to preserve their cultural-historical significance while maintaining, enhancing, and utilizing its core cultural-historical qualities. Rather than halting all development, future spatial

transformations in the area must take this significance into account (Rijksdienst voor het Cultureel Erfgoed, 2024). In the Netherlands, 472 of such protected areas were assigned since 1962 and as of 2012, no more will be added. The Environment and Planning Act (*Omgevingswet*) contains instructions for municipalities about the protection of conservation areas (IPLO, n.d.). In addition to just protecting the conservation areas, sight lines and open areas must also be included as the special character of the areas often extends beyond the delimited area (Rijksdienst Voor het Cultureel Erfgoed, 2024). Other key measures include:

- Preventing activities or developments that could damage or alter the character of protected urban or village views and cultural landscapes, including green spaces and water structures.
- Maintaining the distinctive features and cohesion of these areas.
- Enhancing the quality of public spaces surrounding protected areas to improve their appreciation and preservation.

## 2.7 Monumental status

In addition to entire areas that might be assigned a protected status in the form of a conservation area, individual protection of buildings may also occur in the Netherlands when it is of great importance. Generally, three types of individual monumental statuses are distinguished for buildings that are relevant for this research: national monuments, provincial monuments, and municipal monuments (Monumenten.nl, 2024).

### 2.7.1 National monuments

National monuments are built objects or archaeological sites of national importance that must be preserved by law. This status is assigned to buildings that are important because of their beauty, cultural-historical value or scientific significance (Rijksoverheid, 2022). Approximately 63,000 of such national monuments exist in the Netherlands. The Cultural Heritage Agency of the Netherlands designates the status of national monuments on behalf of the Minister of Education, Culture and Science. Generally, municipalities grant permits for modifications to buildings with a national monumental status.

### 2.7.2 Provincial monuments

Provincial monuments refer to build objects that are designated with a monumental status by the provinces. The municipality is accountable for the maintenance of these objects and issues permits for the adaptation of monuments. Despite that all provinces may designate objects and structures as provincial monuments, only the provinces of North Holland and Drenthe have done so (e.g. many historical farms in Drenthe are assigned with a provincial monumental status).

### 2.7.3 Municipal monuments

Finally, municipal monuments are objects of importance to the municipal identity where the municipality itself regulates the designation of protected statuses (Monumenten.nl, 2022). This designation is often executed by the executive board of a municipality (*college van burgemeester en wethouders*). Like in national- and provincial monumental status, a municipal definition of a monument typically is typically based on attributes such as beauty, cultural-historical value, architectural-historical value, urban development significance, or scientific importance. A municipal monumental status regulates that for activities like renovation, relocation or demolition of a municipal monument, a permit application from the municipality is required (IPLO, n.d.).

### 3. Proposed solutions

---

In the methodology chapter, the case study area is first introduced, followed by a discussion of the georeferencing method. Next, the building vectorization process is explained, then the datasets used in the research are presented. The chapter continues with the techniques used for object matching between temporal versions of buildings, concluding with an overview of the change detection phase. An overview of the proposed solutions for the building vectorization process, and object matching and change detection process is provided in Figure 7 for clarification.

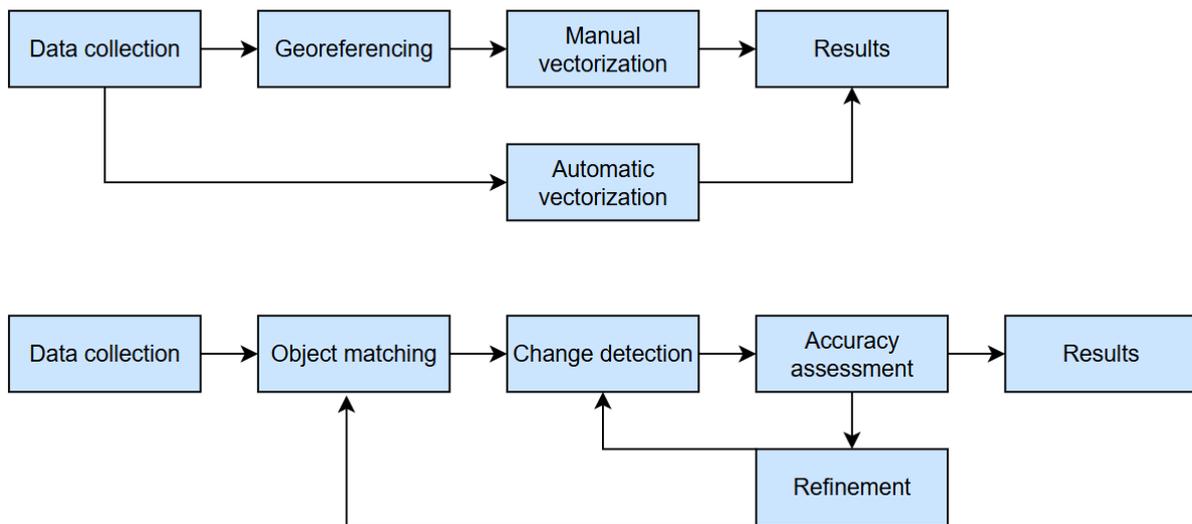


Figure 7: Graphic Illustration of building vectorization from the 1965 cadastral map (above) and object matching with change detection in BGT Pand (below).

#### 3.1 Case study area

The now city of Utrecht was founded by Roman soldiers in 47 AD, when they build a castellum on the south bank of the river Rhine (Dolfin et al., 1989). At the location of what is now the Dom Square, the Romans built a fortification as part of the Lower Germanic Limes, which marked the northern boundary of the Roman Empire. The castellum was given the name Traiectum, which translates to crossing place as the fortress protected a crossable section of the river Rhine. After the fall of the Roman Empire and the subsequent Merovingian Empire in the seventh century, the city was captured by the pagan Frisians. When Willibrord arrived from England in 690 to begin his missionary work, Utrecht that had been recaptured from the Frisians, became base of the bishop's operations. Due to the invasions of the Normans, the bishop of Utrecht ended up in Deventer. It would take until 922 before a bishop would return to Utrecht and the city developed further. In that year, bishop Balderic (897-975) decided to establish the episcopal see again in the Utrecht castellum. Balderic's successive bishops were appointed by the king and later by the holy Roman Emperor, mostly for their administrative qualities, and large amounts of lands were gifted to them. Their rule created a market where traders and producers could sell their goods. These traders and producers started to built dwellings to the northwest of the episcopal residence in the former castellum. In its turn, the creation of a market resulted in a population increase from the 10th century onwards, further accelerating the development of the city. The wealth resulting from Utrecht's development was epitomized by the construction of a large new cathedral by Bishop Adelbold II (975-1026). Even more ambitious was the plan of Bishop Bernold (?-1054), who built a cross of churches around the cathedral after the example of Rome. Immunities (physically closed off, spiritual jurisdictions) were established around these churches, which would partly determine the topography and traffic situation in the later city (Dolfin et al., 1989). In the

following centuries, the city developed further with more civilian settlements being build in the Stathe area, which was situated at the current *Donkerstraat* and later developed towards the *Steenweg*, *Boterstraat* and *Oudegracht*. In 1122, the city of Utrecht was granted city rights by Emperor Henry, which allowed the construction of city walls (Het Utrechts Archief, n.d.). The walled city in the twelfth century retains a similar area and shape to what it is today, as shown in Appendix B.

The city centre of Utrecht - that is the area within the *Singel* (waterway that encloses the historical city centre), excluding *Hoog Catharijne* and *Wijk C*, was named a conservation area on the third of December 1975. This area is illustrated in the left section of Figure 8. The argumentation given for granting the area a protected status is because the historical layout of canals and streets dating mostly from the Middle Ages, along with the surrounding buildings, exhibit such a degree of cohesion and historical significance that the definition of a conservation area in the Monuments Act was deemed applicable (Gemeente Utrecht, 1975).

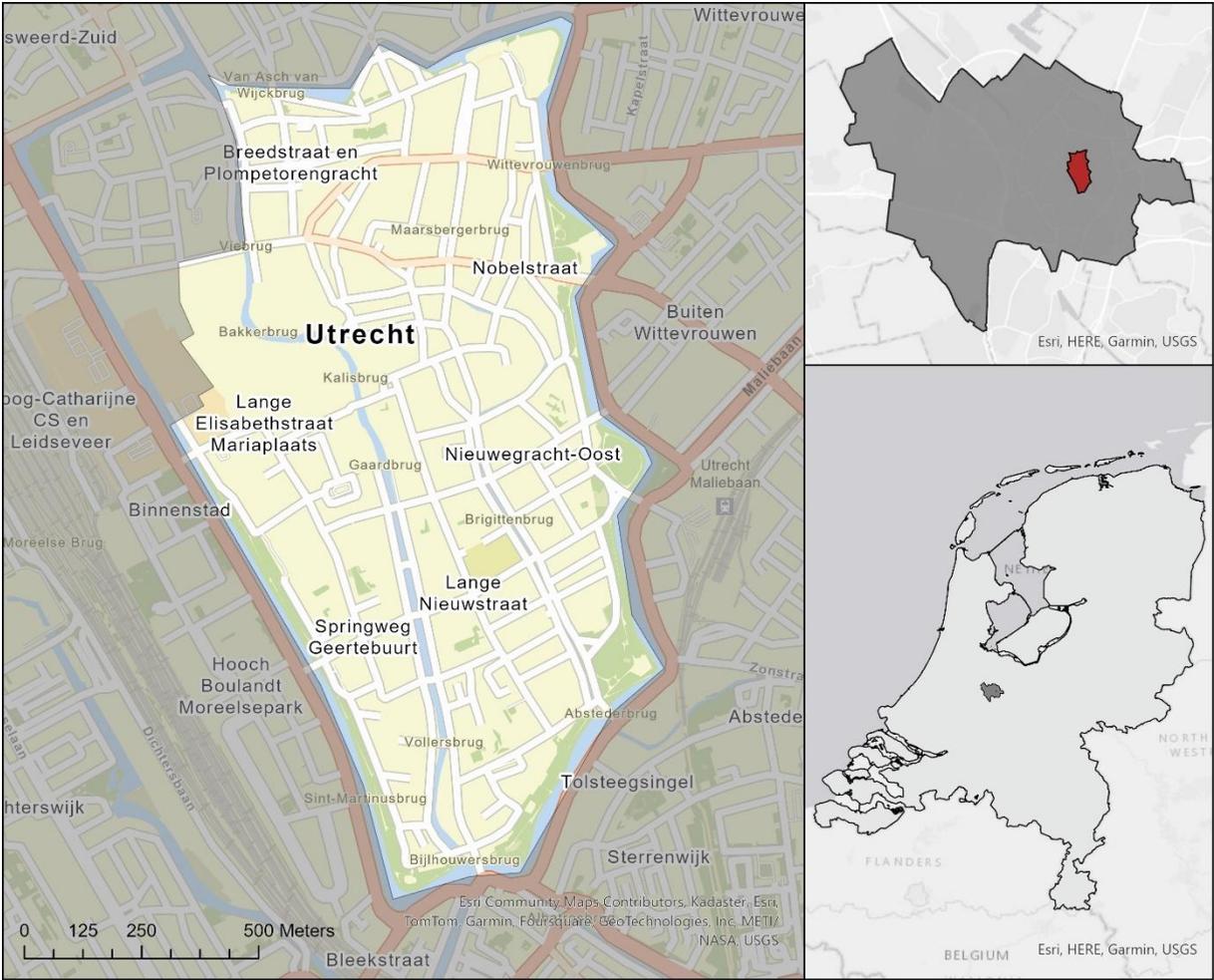


Figure 8: Research area

### 3.2 Data availability

For the process of vectorizing buildings in the conservation area of the historical centre of Utrecht, historic cadastral maps will be used. Cadastral maps primarily focused on representing property boundaries and land parcels, often including details like parcel numbers, ownership, and precise boundaries (Kadaster, n.d.-a). They are essential devices for legal and administrative uses, such as land valuation and property management.

This thesis benefits from access to numerous historical topographic maps, ground plans and cadastral maps of the city of Utrecht that have been digitized through scanning the original paper documents. Such historical sources are made available in relatively high quality in the online archives of organisations such as the Utrecht municipal archives and the Rijksdienst Cultureel Erfgoed (Het Utrechts Archief, n.d.; Rijksdienst Cultureel Erfgoed, n.d.). A particularly useful website for data on the building level is documentatie.org (UDS, n.d.). This project is an informal collaboration between the Municipality of Utrecht, the Utrecht Monuments Fund, the Foundation for Architectural History of the Netherlands, and the Gelderland Society. It provides a comprehensive repository of information on monument conservation, architectural history, and cultural heritage, with a particular emphasis on Utrecht. The site features detailed data on buildings within the city's conservation area and offers an interactive panorama of Utrecht around 1870, viewed from the Dom Tower. Through this panorama, users can access extensive information on individual buildings, including historical photographs, register sheets, and construction drawings. Additionally, this information can be retrieved per plot using the 1832 cadastral map.

While the use of all available datasets is not essential for the purpose of this thesis, documenting these sources provides a valuable foundation for future research. By creating an overview of the various data sources, subsequent studies can build upon this work without the need to independently identify and collect these materials. An overview of the available historical sources of the city of Utrecht are listed in Table 2 with the aim to facilitate further investigations into the historical development of Utrecht's built environment. The large-scale cadastral map of the city centre of Utrecht from 1965 serves as the foundation for the vectorization section of this research and is illustrated in Appendix C. This choice was influenced by the sunk costs associated with the initial analysis and preparation of the dataset, making it a practical and logical focus for the current research. The map contains the buildings in this area, including the cadastral parcels, street names and bodies of water.

Table 2: Historical data sources. The 1965 cadastral map in bold is used for the vectorization process.

Source name	Publisher	Data type	Year	LOD
<b>217122<sup>1</sup></b>	<b>The National Cadastre</b>	<b>Cadastral map</b>	<b>1965</b>	<b>Individual buildings</b>
217121	The National Cadastre	Cadastral map	1955	Individual buildings
RCE-647_Stapper53 <sup>2</sup>	Rijksdienst voor het Cultureel Erfgoed	Inventory of conservation areas	1953	Individual buildings
214055	J. van Druuten	Ground map (plattegrond)	1896	Building blocks
818092 - 81096	The National Cadastre	Cadastral map	1888	Individual buildings
216861	The Municipality of Utrecht	Ground map (plattegrond)	1870	Building blocks
MIN06075A01 - MIN06075C02 <sup>3</sup>	The National Cadastre	Cadastral map	1832	Individual buildings
216732 <sup>4</sup>	The Municipality of Utrecht	Ground map (plattegrond)	1822	Building blocks

<sup>1</sup> <https://hetutrechtsarchief.nl/beeldmateriaal/?mode=gallery&view=horizontal&q=217122&page=1&reverse=0>

<sup>2</sup> [https://beeldbank.cultureelerfgoed.nl/rce-mediabank/detail/8f1c4b7f-d5b7-1b7d-9307-f73603bba2ec/media/98760169-b08e-0ae6-79e1-50de6b4b622e?mode=detail&view=horizontal&q=RCE-647\\_Stapper53&rows=1&page=1](https://beeldbank.cultureelerfgoed.nl/rce-mediabank/detail/8f1c4b7f-d5b7-1b7d-9307-f73603bba2ec/media/98760169-b08e-0ae6-79e1-50de6b4b622e?mode=detail&view=horizontal&q=RCE-647_Stapper53&rows=1&page=1)

<sup>3</sup> [https://beeldbank.cultureelerfgoed.nl/rce-mediabank/detail/a7662e8e-94d7-11e5-a9ca-fb3fe39144dd/media/a4e6167e-6293-ce91-393c-e7833e3d8088?mode=detail&view=horizontal&q=Kadastrale%20kaarten%201811-1832&rows=1&page=189&fq%5B%5D=search\\_s\\_monuments\\_monument\\_county:%22Utrecht%22](https://beeldbank.cultureelerfgoed.nl/rce-mediabank/detail/a7662e8e-94d7-11e5-a9ca-fb3fe39144dd/media/a4e6167e-6293-ce91-393c-e7833e3d8088?mode=detail&view=horizontal&q=Kadastrale%20kaarten%201811-1832&rows=1&page=189&fq%5B%5D=search_s_monuments_monument_county:%22Utrecht%22)

<sup>4</sup> [https://hetutrechtsarchief.nl/onderzoek/resultaten/archieven?mivast=39&mizig=210&miadt=39&miview=inv2&milang=nl&micode=BEELDBANK\\_CART\\_DOC&minr=41627796&miaet=14](https://hetutrechtsarchief.nl/onderzoek/resultaten/archieven?mivast=39&mizig=210&miadt=39&miview=inv2&milang=nl&micode=BEELDBANK_CART_DOC&minr=41627796&miaet=14)

### 3.3 Building vectorization

Building vectorization begins with the georeferencing of the historical cadastral map, which aligns the map to a spatial reference system. Following this, the buildings are derived using a combination of manual and automatic techniques, which will be discussed in the following sections.

#### 3.3.1 Georeferencing

The first step in analysing the historical development of the conservation area involves overlaying historical data onto the city centre of Utrecht. This is accomplished through employing georeferencing techniques, which allow for precise alignment of the input spatial information through determining the position of input data in a spatial coordinate system other than its own (Cascón-Katchadourian & Alberich-Pascual, 2021). Georeferencing a historical map in GIS begins with the digitization of the source. Then, locations must be identified that exist on both the non-georeferenced and georeferenced maps, these are known as ground control points, or GCPs. One should consider that the selected GCPs must have remained consistent over time, objects such as landforms, monuments, or streets are therefore recommended. The process of assigning GCPs to the historical sources and the reference map enables the georeferencing software to align those points at the same coordinates. This method is particularly effective for georeferencing aerial images and maps created in the 19<sup>th</sup> and 20<sup>th</sup> century, which tend to be more geometrically accurate than the less precise and more schematic maps from earlier centuries (Cascón-Katchadourian & Alberich-Pascual, 2021). For georeferencing the historical sources, the 'georeferencing tool' is used within the ArcGIS Pro software. This tool allows for the alignment of a raster dataset to the correct geographic location on the reference map. Furthermore, the software enables the transformation of the source data to a coordinate system. Within the ArcGIS Pro software, three different types of transformations are most widely used to transform the raster dataset to the map coordinates. These are the first-, second- and third order polynomial transformations. The polynomial transformation uses a polynomial built on control points and a least-squares fitting algorithm and are optimized for global accuracy (Esri, n.d.-b). The difference between the three types of transformations is that the higher the order the more complex the distortion that can be corrected. Generally, if a raster dataset needs to be stretched, scaled, and rotated, a first-order transformation is recommended while if it must be bent or curved, a second- or third-order transformation is more suitable.

To assess the accuracy and precision of the georeferencing process, the root mean squared error (RMSE) is an often-used metric in historical maps (Brovelli & Minghini, 2012; Cascón-Katchadourian & Alberich-Pascual, 2021; Baiocchi et al., 2013; Szypuła, 2019). The RMSE is expressed as:

*Equation 1: RMSE*

$$\text{RMSE} = \sqrt{\frac{1}{n} \sum_{i=1}^n (Y_i - Y_j)^2}$$

Where  $Y_i$  are the observed (true) values,  $Y_j$  are the predicted values and  $n$  is the total number of observations. RMSE measures the average difference between the true coordinates which are based on the known reference points, and the predicted coordinates after the georeferencing process. A lower RMSE indicates a higher accuracy, meaning that the historical map aligns closely with the reference map, while a higher RMSE suggests greater distortion. The RMSE is calculated in the same units as the map's coordinates. In the ArcGIS Pro software, three different RMSE residuals are calculated; the forward-, inverse- and the forward-inverse residual. The first of the three shows the error in the same units as the data frame spatial reference, the second indicates the error in the pixel

units and the third measures the accuracy in pixels (Esri, n.d.-b). Typically, only the forward RMSE is interpreted when discussing the georeferencing accuracy.

### 3.3.2 Manual vectorization

A share of the buildings that are derived from the digitized cadastral map are vectorized using manual techniques. Before drawing the polygons, a convolution function is used in the software for sharpening the input raster image using a kernel-based equation (Esri, n.d.-c). This image enhancement tool makes it considerably easier to distinguish lines in the original scan. Then, the buildings are vectorized by manually drawing the vertices of building polygons over the source data using the create polygon feature in the ArcGIS Pro software. This tool allows the user to create irregular polygons comprising unequal sides and angles, which is useful when drawing irregularly shaped buildings (Esri, n.d.-d). As was previously discussed in Paragraph 2.1.1, manual vectorization is a very resource-intensive process. This is also encountered in this thesis, with the vectorization of 100 buildings requiring approximately 20 minutes. Considering that the cadastral map of 1965 used in this research contains some 6000 buildings, the total vectorization process takes roughly 20 hours to complete. Therefore, only a smaller subsection of the input map is digitized by hand. In addition to being an accurate method for vectorizing the data, the manually derived polygons can be utilized as training data when employing a deep neural network approach for the automatic detection of lines in input historical cadastral maps. This is a useful approach as one of the main hurdles for using DNNs for historic cadastral plan vectorization is the limited availability of training data (Ignjatić et al., 2018). Additionally, when training data is available it is not always applicable to other input data, leading to significant degradation of model performance (Oliveira et al., 2017). This process is further discussed in the following section.

### 3.3.3 Automatic vectorization

As discussed in previous chapters, CNN-based approaches are increasingly being explored for automatic map vectorization. While implementing this approach was considered for this research, it was ultimately decided to be unfeasible due to time constraints. However, steps were taken to prepare for potential future work in this direction through the creation of a training dataset by the partial manual vectorization of the 1965 cadastral map. This data can be used for training CNNs or other machine learning methods. This preparation ensures that future studies can build on the groundwork laid here to leverage the advantages of using AI-based models for map vectorization.

In addition to creating training data, an untrained model was also tested for the purpose of map vectorization. The platform tested for this is the so-called Vectorization and Coupling Tool for Reconstruction, or VeCTOR in short. This model was developed by Kadaster for converting JPEG pictures of field sketches (an example of which is illustrated in Appendix D) to digital vectorized networks of geometric observations, coupled to other sketches at overlapping points (Franken et al., 2021). VeCTOR is comprised of multiple pipeline-stages for the vectorization of the input data, in which different artificial intelligence algorithms are used. The employed AI-algorithms, in combination with human validation and feedback allow for the model to perform better with each iteration. The training dataset of the 1965 cadastral map can be used to train the VeCTOR pipeline to better identify the line segments in historical cadastral maps. To train the model for different types of input data, a supervised learning approach is needed. The model relies on pairs of input images in RGB format and corresponding binary masks to learn the distinction between the relevant building outlines and irrelevant background elements or text. The binary masks serve as the ground truth for the corresponding RGB images, which are always inputted in pairs. In the binary masks, the pixel values are categorized into two classes: black pixels represent areas not corresponding to building outlines, while white pixels represent building

outlines. The complete pipeline used in the original consists of 6 steps, as shown in Figure 9. These steps will be shortly discussed in the following sections.

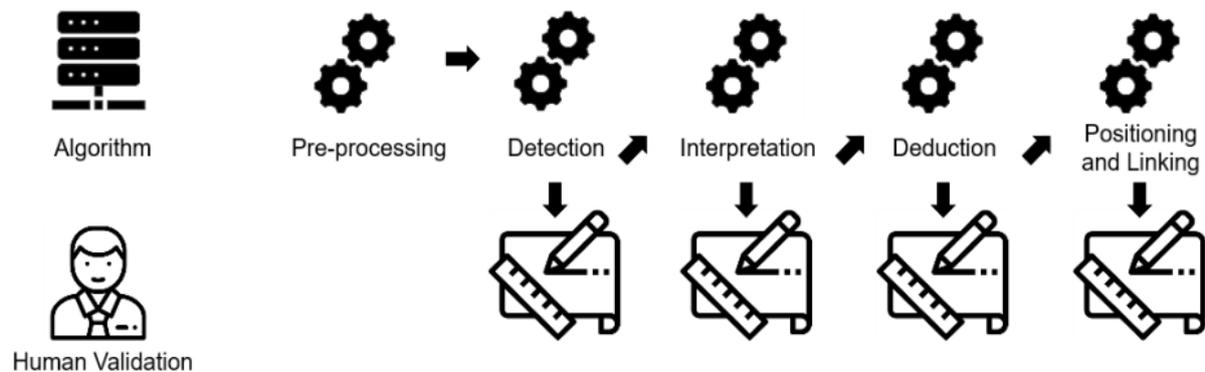


Figure 9: Global pipeline stages of the VeCTOR model. Franken et al., 2021.

### Preprocessing

The original input field sketches of Kadaster are only available in JPEG format, which introduces compression artifacts reducing image clarity and hindering the performance of the model. Therefore, the pre-processing stage accounts for removing noise and irregularities caused by the image compression. In this stage, an S-NET CNN architecture is used based on four residual blocks, comprised of multiple hidden layers, that balance model quality and speed for improving image quality.

### Detection

In the detection step, line and point detection is performed using a combination of LSD (Line Segment Detector) and RANSAC (Random Sample Consensus) algorithms. LSD detects line segments with high accuracy and minimal parameter tuning, while RANSAC constructs continuous lines by linking detected segments. Then, parcel and measurement detection (indicating distances on sketches) is executed, which is the second and final step of the detection stage. Here, a Mask Regional CNN model is employed, combining convolutional layers with additional layers to effectively recognize object regions in images based on bounding boxes and object masks.

### Interpretation

In the interpretation stage, the obtained parcels and measurements are interpreted, and cadastral correction is executed. For interpreting the parcels and measurements, the handwriting of the latter must be recognized, which is a challenging task as they differ a lot. A segmentation-based method is used in a CNN, achieving a word-level accuracy of 40%. Therefore, this step especially requires human validation. In the cadastral correction, the detected and validated measurements and lines are decomposed into measurement lines. The global scale of the sketch is calculated by averaging the scales of all line segments. This scale is used to predict measurement line origins and verify their consistency. The results of this step are presented as overlays on the sketches for user validation.

### Deduction

This stage is tasked with detecting the contours of the building in the field sketch. Candidate building polygons are derived using an approach based on image processing and heuristics, which are then classified as buildings or non-buildings based on properties like background colour and shading. Then, symbols are deducted from the building outlines using the general rule that 180-degree lines indicate walls, and 90-degree angles indicate angles between walls.

## Positioning and linking

Finally, in the positioning and linking stage, the local vectorized field sketches are integrated into a global cadastral map. To perform the positioning step, an edge-matching algorithm is executed, searching for an optimal transformation from local to global geometry. It thereby aligns the sketch edges with reference map edges based on the parcel numbers. The algorithm identifies transformations involving translation and rotation, clustering them using k-Nearest Neighbours to derive optimal transformations via a least-squares approach. Sketches are then automatically linked to other sketches using identified points that are in very close proximity to each other based on a threshold value. This is done to improve the accuracy of the resulting cadastral map.

### 3.4 BGT Pand

The source that will be used in the object matching and change detection phases of this research is the BGT Pand (Basisregistratie Grootchalige Topografie) dataset. It is developed and maintained by the national Cadastre and can be freely accessed through the PDOK website (PDOK, n.d.-a). The BGT is a digital dataset of the Netherlands that uniformly documents objects like buildings, roads, waterways, railway lines, and green spaces (Kadaster, 2024). The use of the BGT is regulated by law, which came into effect in 2016 for source holders and the National Facility (LV BGT). A source holder is the legally designated organization responsible for providing data and are often governments. The purpose of the BGT is to ensure that all government entities utilize a unified base map of large-scale topography for the Netherlands, with governments referring to all levels of government (national, provincial, municipal and water boards) plus other administrative bodies (Geonovum, 2020). Within the government, use is mandatory and based on legislation. Its information is freely accessible to everyone. As mentioned, the Pand (building) object type is used. This is characterized as the smallest functionally and structurally independent unit that is directly and permanently connected to the earth and that can be entered and locked (Geonovum, 2022). Unlike other national databases such as the BAG (Basisregistratie Adressen en Gebouwen), each building within the BGT - including the demolished buildings - have timestamps, enabling change detection (Kadaster, n.d.-b). Historical data is maintained in the BGT to answer questions about data validity and change timing.

#### 3.4.1 Bi-Temporal model

Considering that building versions in the BGT Pand exist within a timeframe, the temporal component must be discussed. When enclosing temporal information about objects within relational databases, two types of times exist. Firstly, the real-world time, which is more formally known as the valid time, and secondly the time that a piece of information enters the database, the transaction time (Thompson & Van Oosterom, 2021; Van Oosterom, 1997). This bi-temporal model has some considerable benefits such as providing more clarity to changes in data over time, which is important when aiming to compare and visualize historical data.

#### 3.4.2 Temporal component in the BGT Pand

As was discussed in the previous paragraph, the BGT Pand dataset includes buildings that currently exist, that have been demolished and that are planned to be constructed. The dataset contains building version history starting from 2016. The BGT Pand data model includes version history through use of the lifespan (*levensduur*) and history (*historie*) of buildings. Both are transaction times, reflecting when the objects change in the database.

- The lifespan of buildings in the BGT Pand indicate the initial creation and expiration of an object in the BGT Pand dataset. It is recorded using the *objectBeginTijd* (object begin time) and

*objectEindtijd* (object end time) attribute fields. When a building is first registered in the BGT, it receives an *objectBegintijd* from the source holder, marking the start of its formal lifespan. If the building is removed from the dataset, it is assigned an *objectEindtijd*. So, while a building is active, the *objectEindtijd* remains null; once it becomes inactive, this field is given a value.

- The history of buildings in the BGT Pand indicate when a change of an object has been made in the registration. This concerns the administrative recording of the object. It is recorded using the *tijdstipRegistratie* (time of registration) and *eindRegistratie* (end of registration) attribute fields. When a building is added to the database, it receives a timestamp from the source holder; the *tijdstipRegistratie*. Then, if a building ceases to exist, for instance due to its demolition, the source holder gives it an end registration date, the *eindRegistratie*. So, when a building is active the *eindRegistratie* remains null; once it becomes inactive, this field is given a value.

When the geometry of an object changes, one of two scenarios occurs: a geometry change while, or a split or merge with one or more other objects. In the former scenario, the object ID is retained, and a new version is created; the current version receives an *eindRegistratie* by the source holder. The source holder creates a new object version. The object retains the same *objectBeginTijd* and is given a new *tijdstipRegistratie*, where *tijdstipRegistratie* is equal to the *eindRegistratie* of the previous version. In the latter scenario, new objects are created, and the old objects receive an *eindRegistratie*.

Status changes to a building must first be observed and reported before being processed in the BGT Pand dataset. This is to ensure that changes are updated to correctly reflect the real-world developments while meeting legal and administrative requirements. All building versions in the BGT Pand dataset get a unique 32-digit identification code, called the object-ID. This ID is determined when the object is created and is retained through the lifespan of the building, even if the object is transferred to another source holder. When a building in the BGT is split or merged, objects are regarded as newly created and are assigned a new unique identification code. Additionally, the unique sixteen-digit BAG identification code is assigned to buildings to ensure interoperability between the two base registrations.

### 3.5 Object matching

The object matching of the different temporal versions of buildings in the BGT Pand dataset is done in the ArcGIS Pro software using different matching techniques. This is done using the ArcPy Python package, the created scripts are illustrated in Appendices E and F. The first step in the process is to extract the old building versions present in the BGT Pand subset of the conservation area of Utrecht from the up-to-date buildings. This is done using a select by attributes operation, selecting the features where the end of registration is not null, and/or their status is plan (see Section 3.4.2). Since these buildings are either old versions of buildings, or ones that have not yet been constructed, they are erased from the dataset. The selection of buildings with an end of registration date represents older versions of buildings, which are used as a reference to identify changes made to buildings within the conservation area since the beginning of the version history in 2017. The dataset created by removing the old and not-yet-constructed selection represents the current scenario and serves as the foundation for detecting changes that have occurred to buildings in the conservation area over time.

After having split the old, current and future versions of buildings in the BGT Pand subset from one another, the object matching can be performed. Since the spatial offsets between buildings between the datasets are relatively small and the data is of similar scale, an overlay analysis is used to match the old building versions to the up-to-date ones. Despite the data being of a similar scale, some buildings

have undergone minor topology changes, i.e. due to topology or geometry corrections or because of simple remeasurements. In some instances, this causes their borders to intersect with (multiple) neighbouring polygons. When using a one to many (1:m) join operation, allowing multiple features to be matched with the target feature, using the intersect match option, this results in multiple false matches to neighbouring buildings. To account for this, a one to one (1:1) join operation is chosen for the initial object matching. The *'have their centre in'* match option is used, which matches features if the centroid (the average of all points in the polygon) of a target feature falls within them. This matching option does not match L-, U or O-shaped buildings as their centres fall outside of the geometry of the buildings. To address these buildings, an additional matching round is conducted using the *'contains'* option. A final matching round is performed using the *'within'* option to link the remaining unmatched building versions from the previous step. This method is preferred over an overlay object matching approach that relies on calculating overlap percentages, as it is easier to implement and provides clearer insights. Furthermore, it avoids the need to create and compute new attribute fields, which would add complexity and clutter the already extensive BGT Pand dataset.

Additionally, the BGT Pand dataset is joined with the BAG dataset, which contains supplementary information. The construction date of buildings, for instance, is not included in the BGT Pand dataset as it only records the transaction time; the date when the building was entered into the database, not the year it was originally built. The additional information provided by the BAG is useful in the subsequent change detection phase for validating identified developments. Buildings are matched using the BAG identification number, which is present in both datasets, allowing for an attribute match. The integrating the BAG-data allows for a more accurate classification, as buildings with a construction date in the BAG dataset after a specified threshold can be classified as newly built. Thereby, the construction date in the BAG eliminates the ambiguity caused by geometry-based methods. In this case, buildings built after 2019 are listed as newly built considering that the PDOK aerial imagery that is used for the verification of the results dates to 2018. This verification will be further discussed in Section 3.7. Demolished buildings are also detected using this attribute matching method. This is done by selecting buildings in the BGT that could not be matched based on their BAG IDs. This occurs because the BAG subset only contains up-to-date buildings, meaning that if a building version in the BGT does not have a corresponding BAG ID in the current BAG dataset, it no longer exists. However, when buildings are merged, this results in the deactivation of their old BAG number, or buildings that were planned but never constructed, may still be incorrectly classified as demolished. To account for this, buildings flagged as demolished that intersect with merged buildings are erased from the change class.

### 3.6 Change Detection

For detecting changes between different versions of building in the BGT dataset, a rule-based approach is used. This technique is chosen for its accuracy and allowing flexible parameter adjustment, as argued in Section 2.5 of the Literature Review. This method can be used because the different versions of the BGT datasets overlap precisely, as they are created and maintained in the same geodatabase. The rule sets are based on the factors that were discussed in the Literature Review and are structured to account for different status changes that can occur to buildings in the dataset. The possible statuses as abstracted from the Literature Review are unchanged, demolished, newly constructed, partly demolished, expanded, merged, and split, characterized by the following criteria:

- **Unchanged:** Buildings with identical geometry and attributes in the old- and new version.
- **Demolished:** Buildings present in the old version but absent in the new version.
- **Newly constructed:** Buildings absent in the old version but present in the new version.
- **Partly Demolished:** Changes in building footprints exceeding a specified negative threshold.

- **Expanded:** Changes in building footprints exceeding a specified positive threshold.
- **Merged:** Multiple buildings in the old version merged into one in the new version.
- **Split:** One building in the old version split into multiple buildings in the new version.

Figure 10 below presents graphical illustration of the different change types, structured as a decision tree to visually guide the classification process.

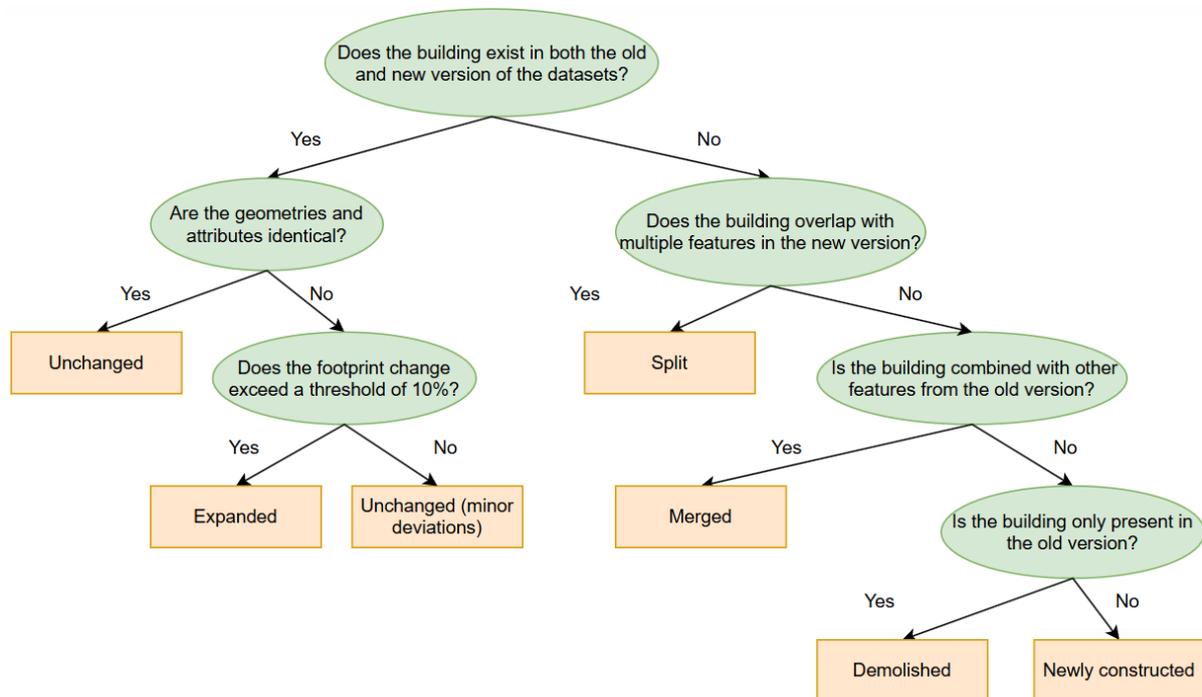


Figure 10: Decision tree for building change detection

### 3.6.1 Area difference ratio

After having matched the old and new buildings versions, a new field called the ADR, short for the area difference ratio, is added to the matched buildings. This field resembles the relative change in area size between the old- and new building versions as linked in the object matching phase. The value can either be negative when the old version is larger than the new one, or positive when the new version is larger than the old one. The ADR is calculated using the calculate field tool with the following formula:

Equation 2: ADR formula

$$(New\ Area - Old\ Area) / Old\ Area$$

The ADR is used to detect the changes that have occurred to the buildings over time, and is coded in ArcPy, as shown in Appendix G. Threshold values are used to categorize the changes in the building areas into the classes that were discussed in the previous section based on the magnitude and the direction of the change in building area.

- An ADR **between -0,1 and 0,1**, meaning that the area of the building has remained unchanged within a margin of 90%, is the threshold for a building to be marked as **Unchanged**. This threshold is chosen to account for potential topology or geometry corrections or remeasurements in the data.

- An ADR **between than -0,1 and -0,3** indicates a moderate decrease in the building's area, suggesting that only a portion of the building has been removed or reduced. Therefore, these buildings are marked as **Partly Demolished**. The upper limit of -0.1 excludes small decreases, which are often geometry changes or remeasurements. The lower limit of -0.3 ensures that larger decreases are also excluded.
- An ADR **between 0,1 and 0,6** indicates a moderate increase in building area, these are therefore classified as **Expanded**. This range captures buildings whose footprint has grown, such as through extensions or additions to the old building versions. The lower limit of 1,1 excludes small increases, while the upper limit omits larger increases that often indicate a merge of buildings.
- An ADR **above 0.6** indicates a large increase in the area of a building, which is often the result of multiple old buildings merging into one new building. Therefore, these buildings are marked as **Merged**. This threshold is chosen to identify large increases, distinguishing them from moderate increase that fall under the Expanded category.
- An ADR **below -0.3** signifies a large decrease in the building's area. This decrease typically occurs when a building is subdivided into smaller parts, resulting in a considerable loss of its original footprint. In these cases, the building is classified as **Split**. The threshold of -0.3 ensures that only large area decreases are included, thereby avoiding moderate decreases of area size.

### 3.6.2 Refinement step

To ensure the accuracy of the change detection model, a refinement step was performed for the identified splits and partial demolitions identified by the initial 1:1 matching process using the ADR as described above. This aimed to confirm that these splits and partial demolitions are indeed valid, rather than partial demolitions, demolitions or other change types. For the split buildings, the refinement step involves selecting the up-to-date BGT buildings that fall within the boundaries of the old split BGT buildings, as attributed by the model. Based on the assumption that splits result in multiple smaller buildings inside the historical one, a one-to-many (1:m) spatial join is performed in ArcGIS Pro between the two datasets using the *'within'* match option. This links each split building to the multiple corresponding updated ones. The number of associated BAG buildings is then counted, and only the BGT buildings with more than one BAG building are selected, as these are the ones that have undergone valid splits. The old building versions with fewer than 2 matches are then transferred to the partly demolished change class, as these have become smaller over time, but do not consist of multiple new buildings. The same process is performed for the partly demolished buildings class by firstly selecting the new BGT versions buildings that have their centre in the old building initially identified as partly demolished. Then, the contain match option is used to join the old partly demolished buildings to the selected new building versions in a 1:m spatial join. The ones that are matched to multiple new buildings that do not share the same BAG IDs are then flagged and added to the Split class. The created ArcPy code for the refinement step is shown in Appendices H and I.

### 3.7 Accuracy assessment

Aerial imagery supplied by PDOK (n.d.-b) is used for the verification of uncertain results in the change detection phase of the BGT Pand data. Uncertain results are identified by manual inspection of the change detection results compared to the old- and current BGT data. The results that are identified as uncertain are then validated using two aerial images in time to confirm the detected changes in the

model. Since 2016, an aerial photograph covering the entire ground space of the Netherlands is published every year. From the 2016 edition, this product is available as open data with a resolution of 25 centimetres. Since the 2021 version, image material with a higher resolution of 7.5 centimetres is also made available as open data through the PDOK portal. As the orthophoto mosaics are available for every year since 2016, and the earliest entries into the BGT Pand dataset also dating back to this year, this allows the detected changes on the building level to be verified through the time. It must be noted that the lower 25-centimetre resolution until the 2021 time step makes it considerably more difficult to recognize changes as opposed to the later 7.5 centimetre resolution time steps. This is especially the case for smaller buildings and buildings that are situated in shaded areas in the pre 2021 aerial images. Verification using aerial imagery is useful for the validation of larger scale changes that can be observed from a top view of the buildings.

The final accuracy of the change detection results is assessed using a confusion matrix, as is similarly employed in the change detection study by Matikainen et al (2012). In a confusion matrix, the performance of classification models can be tested based on the percentage of cases it identifies correctly and incorrectly. This is done in a cross table with the change types of the old version of the BGT in the X-axis, and the new version of the BGT in the Y-axis after. Hereafter, the cases are counted in which the status was identified correctly, which are then entered in the table. When a building is identified incorrectly, this is also recorded, including which incorrect change class the building was attributed by the change detection model. The final output is a table in which the percentage of correctly detected changes are listed per building status.

### 3.8 Historical change detection

Considering the time constraint for this thesis, the object matching and change detection phases are only performed for the different temporal version of buildings in the BGT Pand dataset, and not with large scale cadastral data. However, it is encouraged to use this study as a broader framework for analysing changes using older historical datasets, like the cadastral dataset that was partly vectorized. When wanting to compare the BGT Pand data to the vectorized historical cadastral data, this requires a more complex pipeline similar to that used by Zhou et al. (2018), as discussed in Section 2.5. In this instance, the pipeline must ideally include data matching, accounting for displacement between the historical- and BGT data, shape similarity, morphology of differing components, building pattern constraints and finally a change detection to achieve accurate change detection results.

## 4. Results

In the Results section, the results of the research will be illustrated and shortly discussed. Firstly, the building vectorization processes are considered. Hereafter, the object matching is elaborated on and the chapter concludes with the results of the change detection phase.

### 4.1 Building vectorization

In the building vectorization, the georeferencing of the historical cadastral map and the manual- and automatic vectorization processes are illustrated and briefly discussed.

#### 4.1.1 Georeferencing

The georeferencing of the historical cadastral map of the conservation area in Utrecht was conducted using a third order polynomial transformation. This method was chosen as it accommodates well for non-linear distortions, which are often present in historical map data. In the transformation process, 50 control points were used to ensure that the historical map data would be aligned robustly to the modern coordinate system, as more control points of good quality contribute to a more accurate polynomial transformation. The placement of the GCPs is illustrated with red points in the left pane of Figure 11 below. The final georeferenced map is overlaid over a topographic basemap and is edited using the darken layers blend. This results in the blending of the historical map with the content below it in the basemap to illustrate the accuracy of the georeferencing process, as is shown in the right pane of Figure 11.



Figure 11: GCPs in the research area (left), georeferenced historical map (right)

To assess the accuracy of the georeferencing process, the Root Mean Square Error (RMSE) is used, as is shown in Table X. The forward RMSE indicates an average deviation of 1.41 map units between the GCPs and their position after the transformation.

Table 3: RMSE values of the georeferencing process

Forward RMSE	1.409540
Inverse RMSE	0.014409
Forward-inverse RMSE	0.000242

#### 4.1.2 Manual vectorization

For the manual vectorization of the buildings in the historical cadastral map of 1965, a total of 2,059 buildings were digitized by hand by the researcher and one peer. This process took approximately 7 hours to complete, over the time span of a week. Most of the buildings that were vectorized are situated in the south and west, including some smaller sub sections along the northern parts of the research area. An overview of the manually vectorized buildings is illustrated in Figure 12.



Figure 12: Overview of the manually vectorized buildings of the 1965 cadastral map

In addition to providing serving as a resource for studying historical urban development, the manually vectorized buildings can be used to train a deep learning model, like a CNN-based pipeline to automate the process of building vectorization. To train most of such models, including the previously discussed VeCTOR pipeline, the vectorized buildings must be illustrated in white, with the background in black, creating a binary mask. These masks are then paired with their original RGB counterparts, which in turn allows the model to learn the distinction between the relevant building outlines and irrelevant background elements or text. Two examples of such image pairs are shown in the Figure below. In order to facilitate further research aiming to automatically vectorize historical cadastral map data in a similar manner using deep learning, the manually created building dataset will be published on a dedicated Github page. Through this page, the data can be freely downloaded and used by researchers that might be interested in using it for future studies.



Figure 13: Examples of RGB images and corresponding binary masks

### 4.1.3 Automatic vectorization

In the subsequent automatic vectorization process, the historical data was inputted into the base VeCTOR pipeline. This was done to explore the potential of using an already existing CNN-based model for automatically extracting building outlines in the historical large scale cadastral map of 1965. Multiple subsets of the RGB map were inputted into the model to analyse its effectiveness in vectorizing buildings without training the existing model. Some outputs of the untrained VeCTOR pipeline are shown in Figure 14 below.

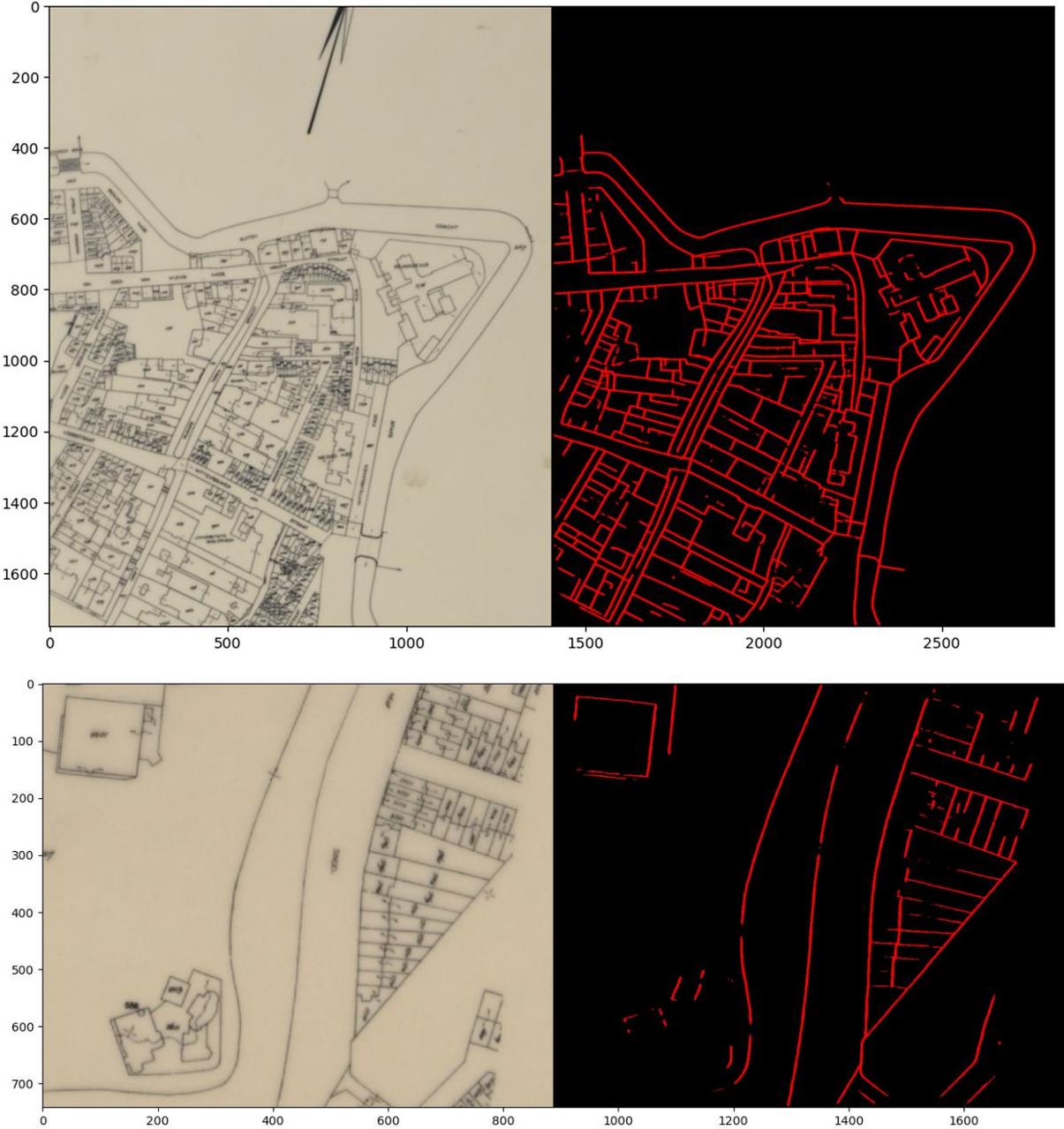


Figure 14: VeCTOR pipeline line segment output

Upon analysis of the source code of the model, the VeCTOR line detection model outputs confidence scores for each pixel. These confidence scores represent the likelihood of pixels belonging to the line class or not. Confidence values range from 0 to 1 and are set at a threshold level in the source code to ensure that pixels are classified correctly according to the set threshold. In the standard configuration of the model, it is set to 0.5. When lowering the confidence level in the source code, the model outputs more line segments as more pixels meet the lowered threshold. This results in the model closing the unclosed line segments more frequently, and the model recognizing more lines overall when inputting subsets of the large scale cadastral maps. Figure 15 shows the differences in the line detection when lowering the confidence level from 0.5 to 0.05, 0.005 and 0.0005. However, lowering the confidence level also causes pixels with a weak probability to be attributed as lines, causing the overall line thickness to increase and potentially causing more false positives to occur in the output. This is especially apparent when analysing the 0.0005 confidence level in Figure 15, which visibly causes the line segments to become thicker than in the outputs with a lower confidence level.

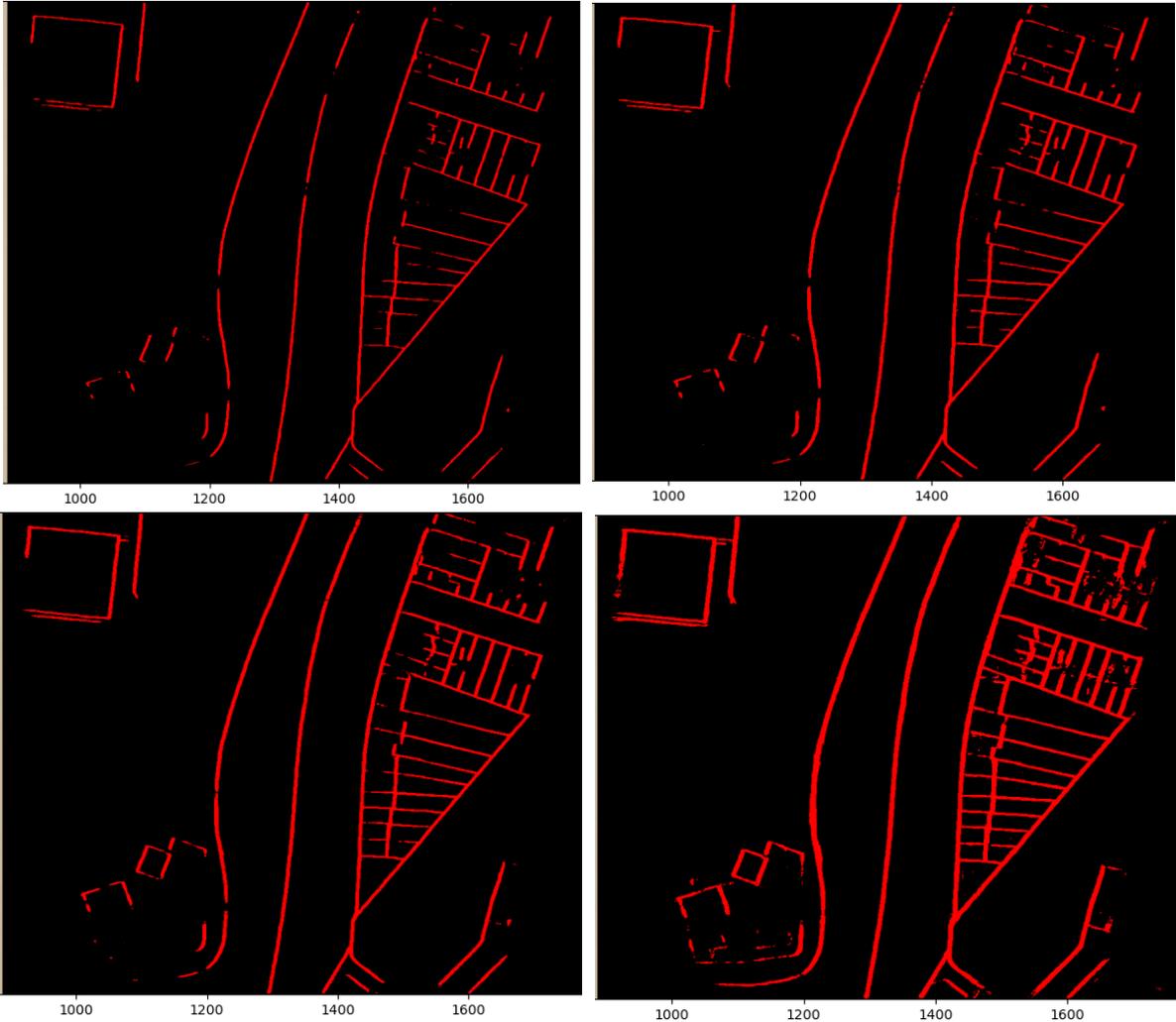


Figure 15: VeCTOR outputs with decreasing confidence levels (0.5 top left, 0.05 top right, 0.005 bottom left, 0.0005 bottom right).

The VeCTOR model has only been trained on upright fieldworks, which influences its performance when applied to different input sources. An implication of the used training data is that the model has learned that lines in the header and footer never need to be vectorized. Additionally, it was trained using fieldworks with a similar scale. Because the inputted cut outs of large scale cadastral maps are without a header and footer and with a different scale and drawing style, the predictions might be incorrect. Another factor that should be considered is that the model's performance is largely dependent on the orientation of the input image. It was found that when the input image was rotated with 90 degrees, this had a large effect on the effectiveness of the model capturing of line segments in the output. This discrepancy is illustrated in Figure 16. Here, the same cutoff of the map is shown, however, in the first image it is rotated correctly, whilst in the second image it is rotated 90 degrees to the left. In the rotated image, the line segments are vectorized much more accurately than in the original image, which only shows fragmented lines in the output.

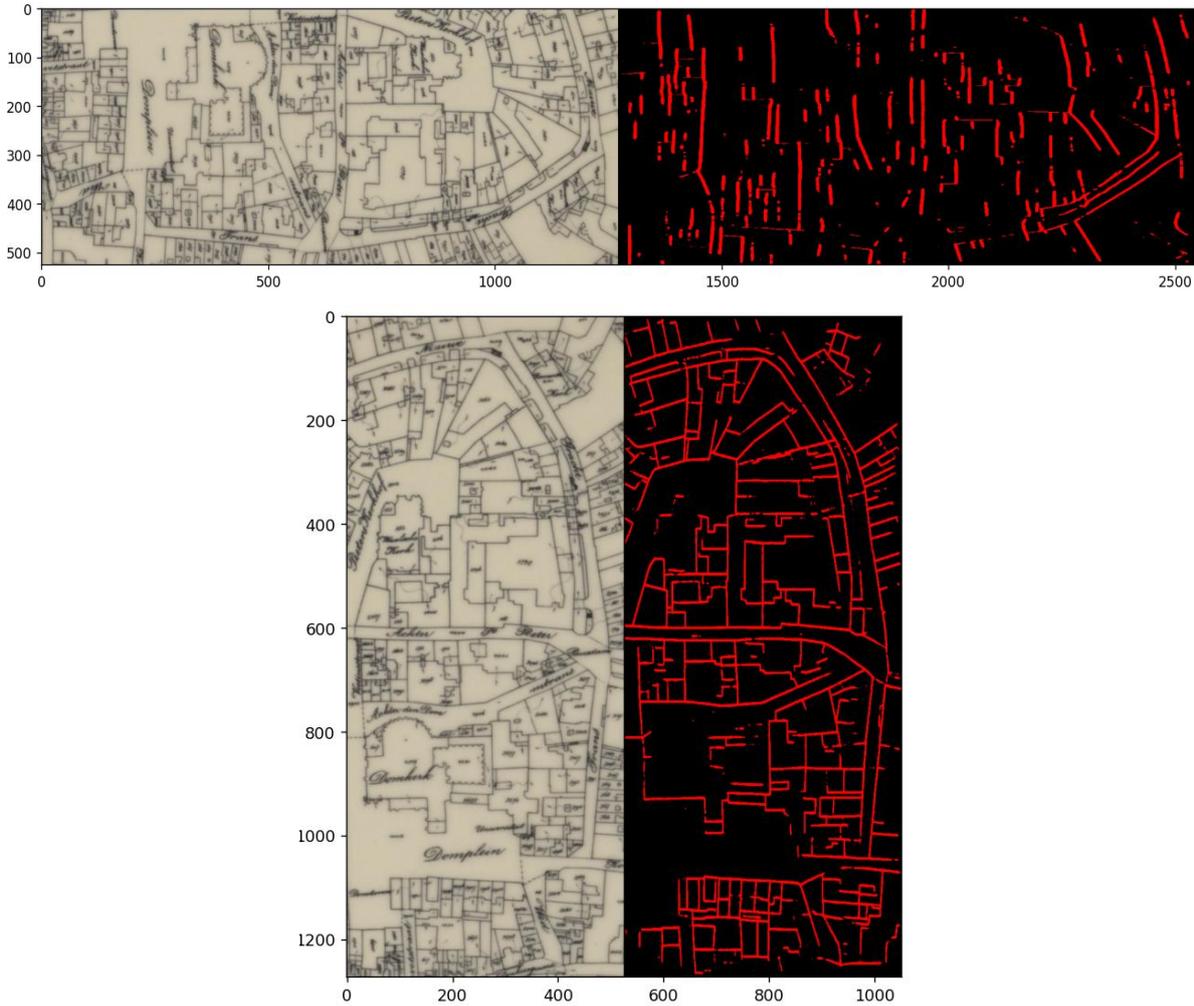


Figure 16: VeCTOR output difference when changing input image orientation

#### 4.4 Object matching

In the object matching step, all historical versions of buildings in the BGT Pand dataset are matched to their updated counterparts, in a total of 3 steps. In the first object matching step, a one-to-one (1:1) join operation using the *'have their centre in'* match option is used, joining old buildings if their centroid falls within a new building's boundaries. This matches 1,581 out of the 1,675 building versions. As shown in light orange in Figure 17. The second matching round is conducted using the *'contains'* option, joining an additional 40 of the 94 previously unmatched buildings. Finally, the *'within'* match option adds six more building matches. Upon manual inspection of the dataset, three out of the 48 unmatched buildings were falsely unmatched. It is decided to remove these buildings from the subsequent change detection phase. The other unmatched buildings have either been (partly) demolished or have been removed from the BGT Pand dataset for another reason, for instance a pergola wrongfully labelled as a building.

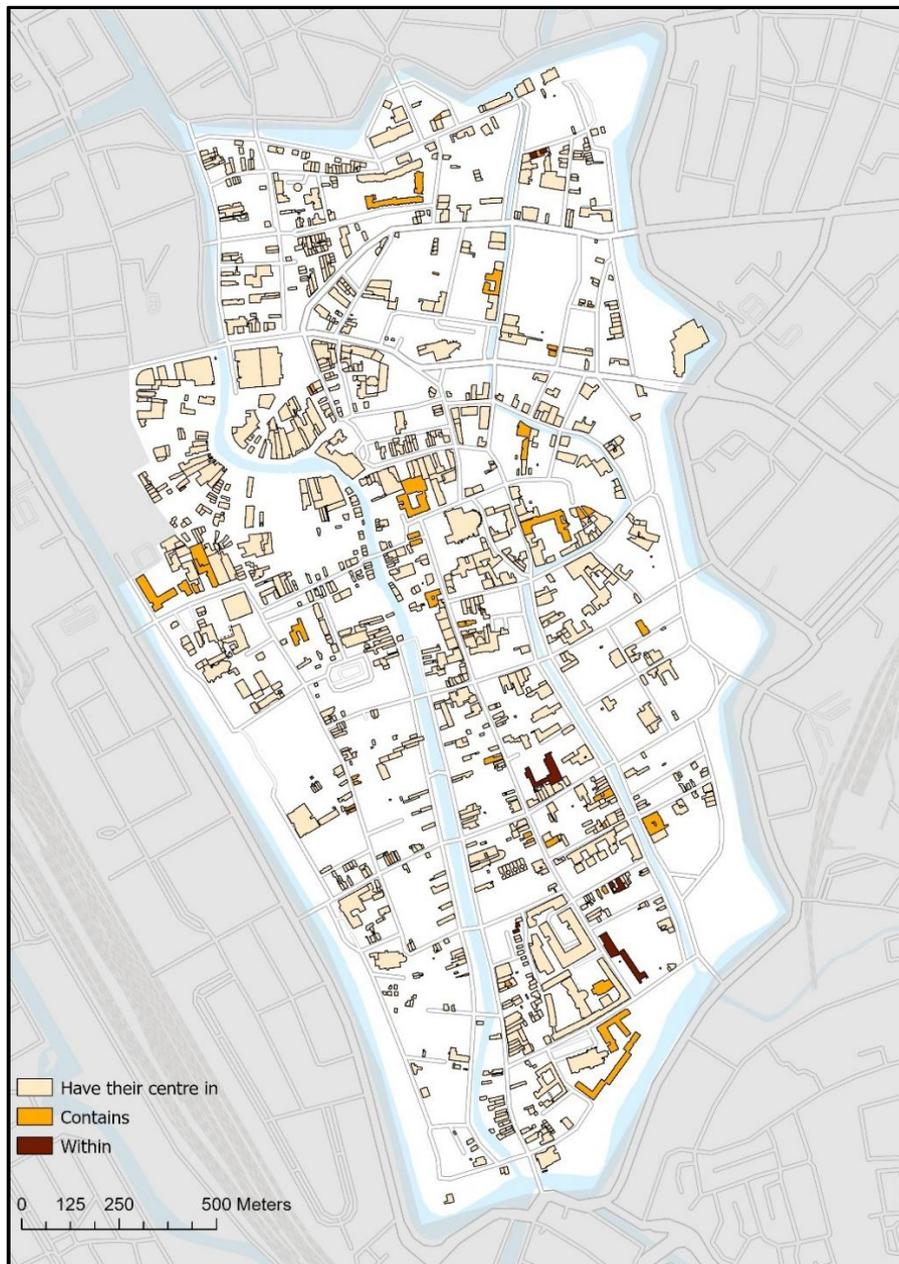


Figure 17: Object matching

4.5 Change detection

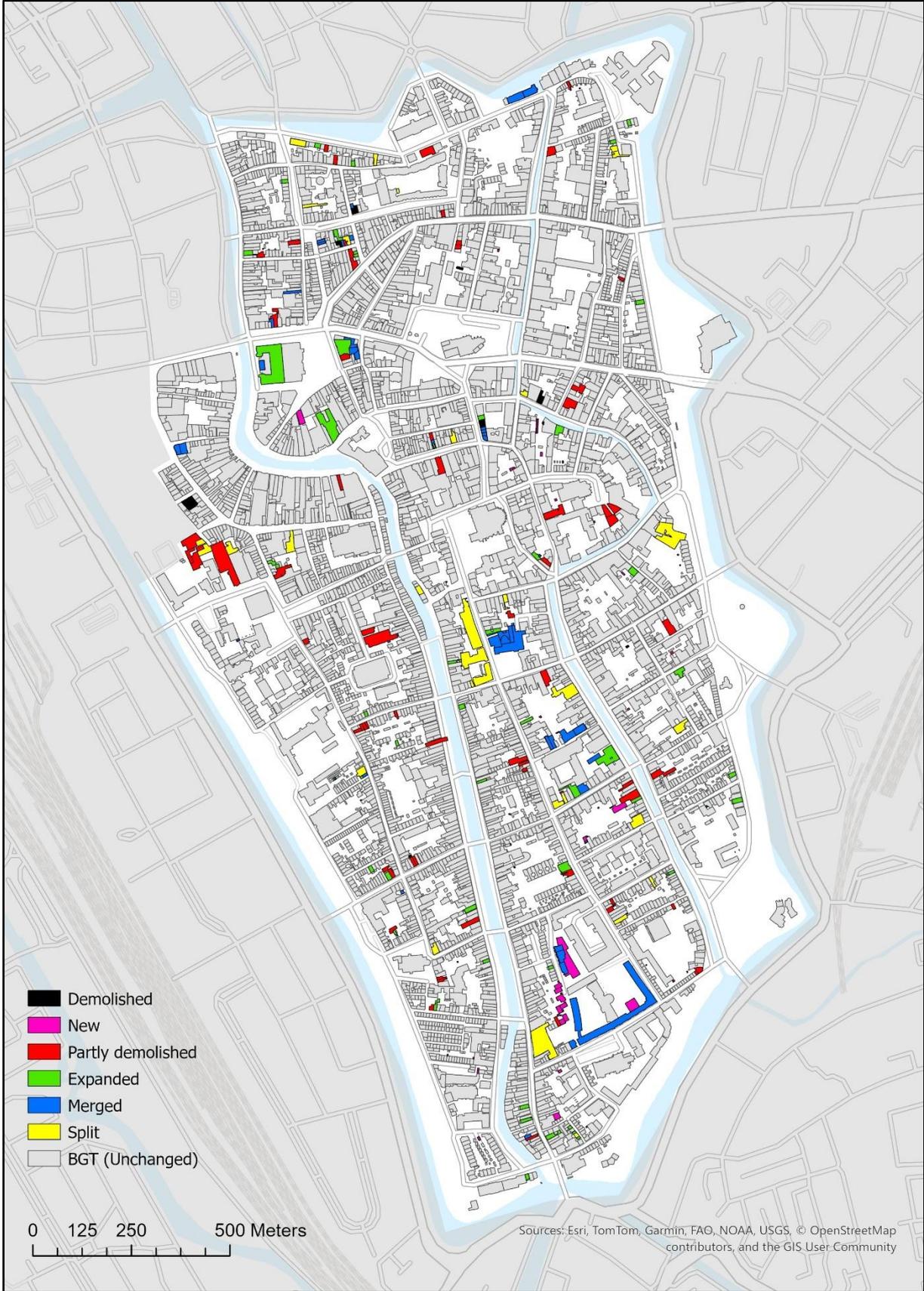


Figure 18: The detected building version changes in the conservation area of Utrecht, between 2016 and 2024

In Figure 18 above, an overview of the final change detection model is shown, with different colours illustrating the different types of changes. The changes between 2016 and 2024 appear to be evenly distributed throughout the conservation area, with no regions standing out due to a higher concentration of urban transformations. In the initial model without refinement, out of the seven possible statuses as identified in Chapter 3, 53 building versions were demolished, 65 were newly constructed, 82 were partly demolished, 78 were expanded, 107 were merged, 41 were split and 4394 remained unchanged. After the refinement step for split building versions, 8 falsely allocated splits were reassigned to the partly demolished class. The refinement of partial demolitions reassigned 8 instances to the split change class, resulting in a total of 42 splits and 82 partial demolitions. A total of 427 building versions underwent change, while 4394 remained unchanged, indicating that 9.7% of the building versions in the conservation area of Utrecht changed in the span of 8 years.

In Table 4, the results that are shown in Figure 18 are illustrated in table form using a confusion matrix, which is also used as accuracy assessment of the model. This confusion matrix shows how many instances of a particular change type were identified successfully by the researched BGT change detection model, as is shown in the bold numbers in the main body of the table. Additionally, the falsely identified cases are presented in the table to study the (un)certainly of the model to accurately recognize the changes that occurred and attribute them to the different change classes. In the table, the 'other' class is added to account for buildings to which multiple changes have occurred, or modifications that are difficult to classify under a single of the identified category. The result shows that overall, the model can accurately identify the changes that occurred, with percentages between 87.2 and 98.5 percent. The refinement step considerably improved the change detection model output, increasing the accuracy of identified splits from 70.7% to 90.4% and partial demolitions from 82.2% to 90.2%. The output of the split and partly demolished refinement step can be seen in Appendix J. It must be noted that the demolished-, new- and unchanged buildings were not manually validated like the other change types, as these were collected using a method based on attribute matching between the BGT Pand and BAG data. Considering that the BAG data is maintained as an official source and is expected to be highly reliable, these are considered to be 95% correct, with an error margin of the residual 5% to account for potential inconsistencies.

Table 4: Confusion matrix of the change detection results

	<i>Demolished</i>	<i>New</i>	<i>Partly demolished</i>	<i>Expanded</i>	<i>Merged</i>	<i>Split</i>	<i>Unchanged</i>	<i>Other</i>	<i>Total</i>	<i>% Correct</i>
<i>Demolished</i>	<b>53</b>	-	-	-	-	-	-	-	53	<b>95.0%</b>
<i>New</i>	-	<b>65</b>	-	-	-	-	-	-	65	<b>95.0%</b>
<i>Partly demolished</i>	-	-	<b>74</b>	-	-	-	-	8	82	<b>90.2%</b>
<i>Expanded</i>	-	-	-	<b>68</b>	-	-	1	9	78	<b>87.2%</b>
<i>Merged</i>	3	-	-	-	<b>100</b>	-	2	2	107	<b>98.5%</b>
<i>Split</i>	-	-	-	-	1	<b>38</b>	-	3	42	<b>90.4%</b>
<i>Unchanged</i>	-	-	-	-	-	-	<b>4394</b>	-	4394	<b>95.0%</b>

The following four figures show examples of change detection instances that were correctly identified by the created model and were validated using PDOK aerial imagery, as discussed in Section 3.5.5. Splits and merges are not illustrated in this manner, as these change types are not visible from the outside of the concerning buildings. Firstly, in Figure 19 an example of a building in the BGT Pand dataset that was correctly classified as demolished shown. In the left pane of the map, aerial imagery dating from 2021 is used as a base on which the demolished building outline is projected. In the right pane, the same location is shown in 2024. Through using aerial imagery of 2021 and 2024, it is illustrated that the building has been demolished.



Figure 19: Demolished building

In Figure 20 below, several buildings that were identified as newly constructed in the southern part of the conservation area are depicted. From the aerial imagery in the left and right map panes dating from 2021 and 2023, it is illustrated that buildings were indeed constructed in this area.

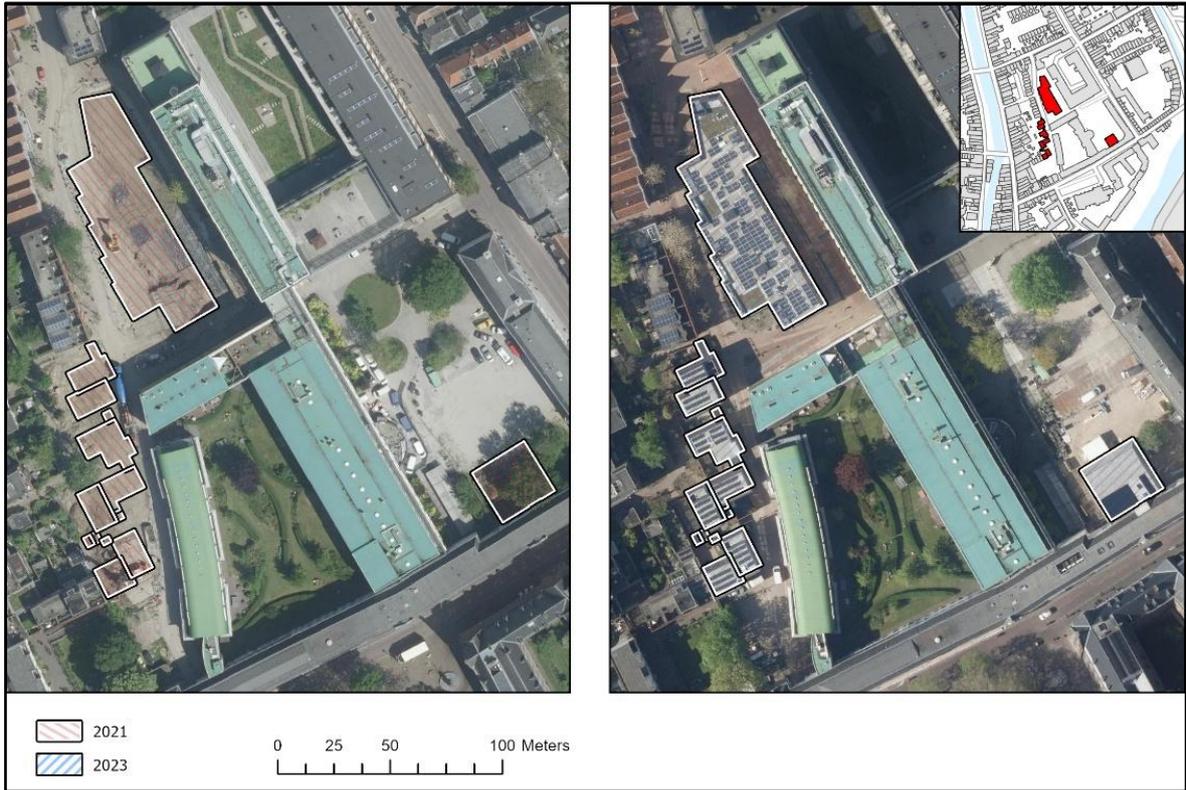


Figure 20: New buildings

Figure 21 is an example of a building identified as partially demolished. It is located next to another building where a new section was constructed, as shown in the previous figure. Here, aerial imagery of 2016 and 2021 reveals that a portion of the building was demolished.

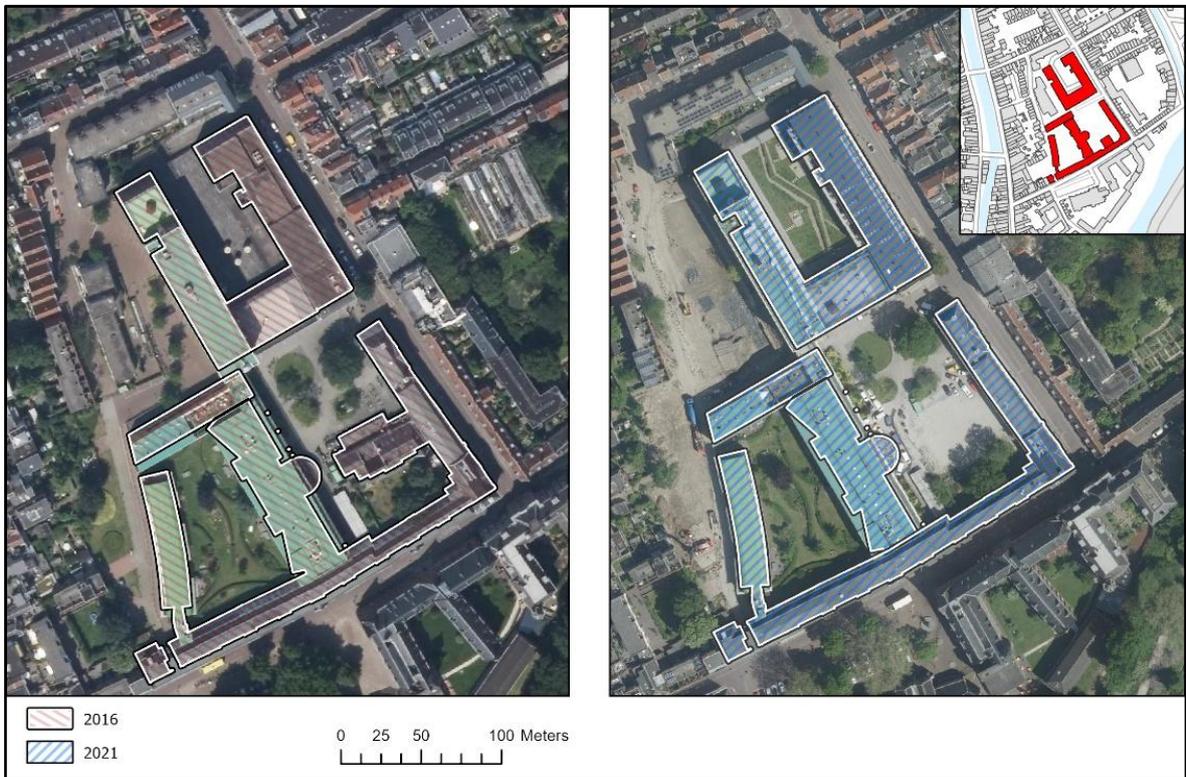


Figure 21: Partly demolished building

Finally, Figure 22 illustrates two buildings in the left map pane that were expanded and merged. The building is classified under both change types because it was originally composed of a smaller and a larger structure, which becomes evident when examining the building outline in the left pane. From the perspective of the smaller building, it was merged, as its relatively small size resulted in a significant change in its ADR. In contrast, the larger building experienced only a slight increase in size due to its already larger area, which is why it was classified as expanded.

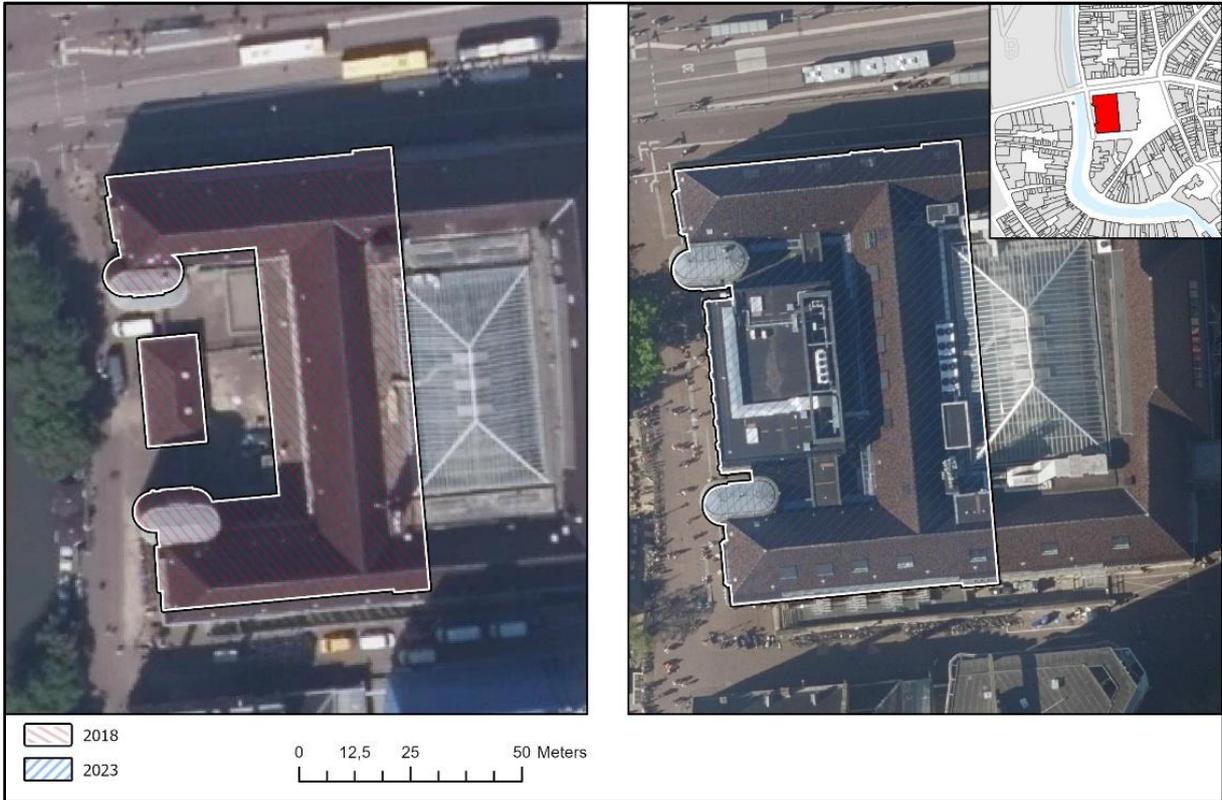


Figure 22: Expanded/merged buildings

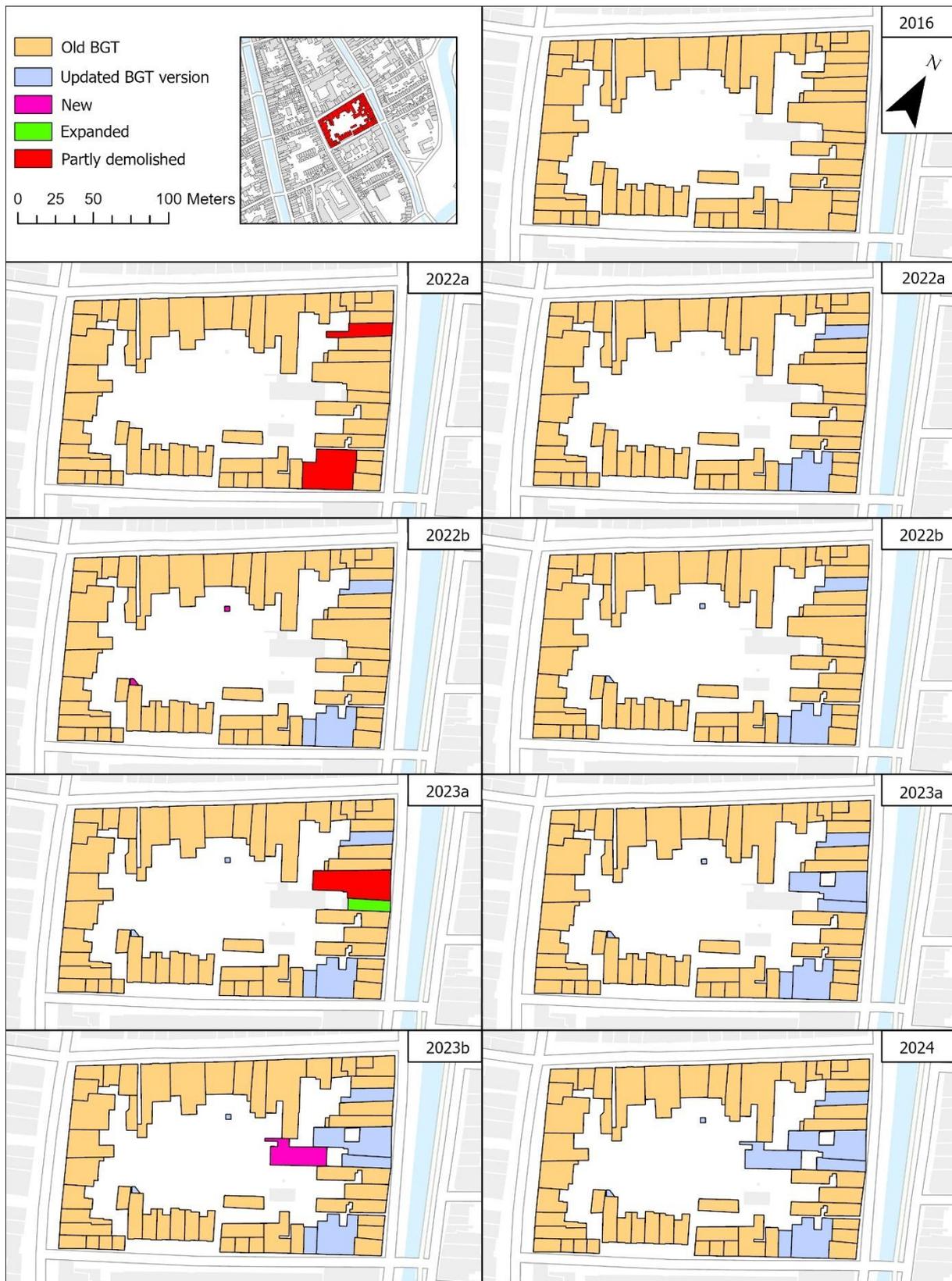


Figure 23: Timelapse of BGT building changes between 2016 and 2024

Figure 23 above illustrates the changes that have occurred to a block of buildings in the BGT Pand dataset in time. The pane in the top right of the figure shows the reference scenario in 2016 without any alterations to the buildings in the BGT Pand having taken place. The transformations are categorized by the year of their occurrence in chronological order, with a maximum of two changes being illustrated per map pane to ensure clarity. Additionally, the old building versions in the left panes that underwent changes are depicted in the colour of the change type that was attributed to them by the change detection model. In the panes on the right, the new version of the buildings are shown in light blue in the way that they are listed in the updated version of the BGT Pand dataset. As can be seen in the figure above, in the 2022a time step, two buildings are partly demolished according to the model. Then, in the subsequent 2022b pane, their updated version are shown. When inspecting the buildings in question, it becomes clear that the one at the top has indeed become smaller. However, the building bottom of the block has not only been partly demolished, it has also been split. This is because building versions can only be attributed to one change class, which will be further elaborated on in the Limitations section. In time step 2022b, two small building that were newly constructed are shown which are thus added to the updated version of the BGT. Then, in the 2023a pane, one building is partly demolished, whilst its direct neighbour is expanded. In the final 2023b time step, a building is newly constructed in the courtyard of the building block, illustrated in pink. This building is then added to the BGT in the lower right map pane.

## 5. Discussion

---

In the Discussion chapter, the outcomes from the Results of the building vectorization, object matching and change detection are interpreted and elaborated on more extensively within the context of this research. The chapter concludes with some limitations of the used methodology.

### 5.1 Building vectorization

In the building vectorization, the georeferencing, manual- and automatic vectorization results are interpreted in more detail.

#### 5.1.1 Georeferencing

The RMSE of 1.41 map units indicates that the transformation captures the overall alignment of the historical map well. However, it also demonstrates that there are inaccuracies in certain areas. These are likely caused by distortions in the original historical map considering it was hand drawn and measured and is thus more error-prone than newer sources. This underlines overall difficulties that are often encountered when doing comparative studies using historical map data, as is also concluded in research by Heitzler and Hurni (2020) into the reconstruction of historical building footprints in the historical Swiss Siegfried map. However, considering that its primary purpose is to serve as potential training data for an automatic vectorization model, localized inaccuracies are inconsequential.

#### 5.1.2 Manual vectorization

When analysing the results of the manual vectorization process, it can be concluded that the results are of overall good quality. The binary vectorized polygons in combination with the source RGB map can be used in further research to effectively train an automatic vectorization model to identify line segments from similar large scale cadastral maps. Nevertheless, it was observed in the process that the quality of the results was dependent on the contributors' abilities and motivation to accurately draw the features in the maps. This is in line with findings presented in research by Chen et al. (2024). Furthermore, it must be noted that in the 1965 large scale cadastral map used as source data, it was at times difficult to distinguish what lines correlated to actual building outlines. In some instances, gardens, parcel boundaries or road segments were drawn very similarly to buildings. Familiarity with the research area facilitated this process.

#### 5.1.3 Automatic vectorization

The results of the automatic vectorization process using the VeCTOR pipeline without training the model on the large-scale cadastral maps were promising. As is shown in Figure 15 in the Results section, the model recognized most of the line segments in the input data, demonstrating strong capabilities in detecting objects in historical maps. This indicates that even with minimal adaptation, the VeCTOR pipeline has the potential to be an effective model for processing large scale cadastral data. Despite these successes, some challenges were also observed. The VeCTOR pipeline does show difficulties in closing the identified line segments, causing many of the lines to be unfinished in the model output. Furthermore, it is observed that building outlines that are overlaid with other shapes, like the building in the bottom-left corner in Figure 15, hinder the model in correctly detecting building outlines. Additionally, fine line segments in larger scale inputs posed a problem for the model, in such instances it was not able to identify the small buildings, as is shown in the top pane of Figure 15.

The observed increase in line thickness when lowering the confidence scores, as illustrated in Figure 16, does not pose a problem for the use of the results in this field of research. This is because processing tools like raster-to-polyline are unaffected by line thickness in raster outputs. Additionally, at the tested confidence levels, small outlines are not yet being completely filled in by the model. However, if this were to happen, for instance in sections with very thin or fine line segments, small buildings could be misclassified as walls rather than individual objects. Further refinement of the confidence threshold may be necessary to balance improved line closure with the risk of false positives.

Finally, the discrepancies seen in the model's performance between different orientations of the input data can be caused by a myriad of factors. One explanation being the cadastral fieldwork data that were used to train the model, which predominantly have horizontal and vertical line orientations. As a result, the model may detect lines more effectively when they align with these orientations. Another factor to consider in is the writing in the input cadastral maps being aligned vertically instead of horizontally. In the original cadastral fieldworks that VeCTOR is trained with, writing on the source data almost exclusively positioned horizontally. However, in the inputted subset of the cadastral map, it is not. This might cause the model to have difficulties in calibrating what pixels it identifies as line segments, and which as text, as it filters out the latter in the interpretation phase of the pipeline.

Due to the phenomenon known as the black box problem, it is impossible to draw conclusions on the internal workings of an AI-based tool based solely on the model's inputs and outputs, as underlined by Guidotti et al. (2018). Therefore, pinpointing a definitive explanation to account for the observed challenges is difficult. However, after consultation with one of the developers of the VeCTOR pipeline, it was determined that the identified challenges in the model can likely be resolved relatively easily by training the model with the manually vectorized building data. This would allow for better recognition of specific line segments in the input data, enabling the model to more accurately vectorize large scale cadastral maps. As previously discussed, implementing this solution was not feasible within the scope of this thesis due to time constraints, however, it does provide a next step for further research.

## 5.2 Object matching

The centroid-based object matching method performed well in matching different temporal versions of buildings in the BGT Pand dataset. Its degree of flexibility helped to correctly match corresponding buildings with small offsets through time, while accounting for potential mismatches caused by their overlaps, which occurred when using the intersect match option. The method does depend on a level of continuity with regard to building topologies. In other words, the general topology between different temporal versions of used datasets should remain relatively consistent and must not shift by more than several centimetres, as a large offset would prevent the successful application of the object matching method. To counter these difficulties, controlled alignment, possibly in combination with pattern recognition can be used, as successfully employed for change detection by Zhang et al. (2013).

Additionally, as outlined in the methodology, the initial object matching approach made use of 1:1 object joins. This was decided as it simplifies the subsequent change detection by preventing incorrect matches caused by topology offsets between different versions of buildings, as also demonstrated by Zhou et al. (2018). While this approach was effective for most change classes, it did cause some challenges for differentiating between splits and partial demolitions. In these cases, multiple historical- or newly created building versions often corresponded to a single counterpart, complicating the matching process and requiring more nuance to the model. Therefore, the additional refinement step

allowing historically split and partly demolished buildings to be joined to multiple new buildings (1:m), significantly enhanced the accuracy and reliability of object matching.

### 5.3 Change detection

Overall, the constructed change detection model that was developed performed well in identifying the occurred changes to building versions in the research area over time. As was mentioned in the Results Section, a total of 9.7% of the building versions in the conservation area of Utrecht changed in the span of 8 years. This demonstrates that urban transformation actively takes place in the studied conservation area, including the demolition of buildings. Thereby, the findings of this research challenge the assumption that protected historical centres in the Netherlands are like frozen, open-air museums where no changes are permitted.

According to the validation of the results, as is shown in the confusion matrix in Table 4, between 87.2 and 98.5 percent of the building version changes were attributed to the correct change class. Furthermore, the refinement step contributed considerably to the improvement of the model, increasing the percentage of correctly identified splits by 19.7 percentage points and correctly identified partial demolitions by 8 percentage points. The confusion matrix indicates that merges were classified most accurately with a percentage of 98.5%, while the expanded class was most prone to inaccuracies, with 87.2% being identified correctly. Most misidentified expansions belonged to the 'other' class, where multiple changes had occurred, or when modifications were difficult to classify under a single of the identified categories. This misclassification arose because in such instances of buildings being incorrectly marked as expanded, they had in reality undergone entirely different shape configurations. This makes it difficult to check whether the building versions had truly expansion or had changed in another way. This was strengthened by the fact that many identified expansions were not visible in aerial images, likely due to discrepancies between data entries in the BGT Pand dataset and corresponding real-world structures, a point that will be elaborated on in the next paragraph.

An unexpected insight into the transformations within the conservation area is that partial demolitions, following merges, are the most common changes observed between 2016 and 2024. This is surprising as there are strict instructions for the modification of buildings in a conservation area as to not damage or alter its special character, as described by the *Rijksdienst voor het Cultureel Erfgoed* (2024). However, when validating the identified changes with PDOK aerial imagery, it becomes apparent that a portion of the identified partial demolitions do not correspond to actual physical changes in the buildings. Instead, these changes can often be attributed to adjustments made in the BGT Pand data model for technical or classification reasons. For instance, structures that were initially classified as buildings, or parts of buildings that are not on ground level, might be re-evaluated and removed from updated versions of the BGT Pand dataset. As a result, a considerable amount of the partial demolitions in the model show changes in the dataset's classification, instead of actual physical transformations of buildings. Such discrepancies between buildings in databases versus their real-world counterparts are in line with the findings of Zou et al. (2018), who discuss that changes in databases do not always represent physical changes and the challenges this introduces for change detection. This emphasises both the complexity of working with large-scale building data and the importance of considering physical changes as well as data updates when analysing changes in building databases.

Additionally, it was found that relying solely on the ADR for the change detection was too simplistic, making the refinement step also essential for improving the accuracy of the change detection model.

Here, the manual verification process played a critical role, as it proved the necessity for a more elaborate change detection pipeline. This conclusion was drawn through validation against the up-to-date BGT Pand and BAG data, in addition to the used PDOK aerial imagery. Thereby ensuring that when including the refinement step, the results aligned more accurately with the real-world changes.

Modelling urban transformation over time in a historically significant area enables a deeper analysis of detected changes, offering insights into the driving factors behind them. Furthermore, the proposed method supports data-driven decision-making by offering a historical baseline for understanding urban transformations, which is crucial for maintaining the character of conservation areas alike. Since the method was developed for the BGT Pand dataset in general, it can also be applied to other cities and towns in the Netherlands to identify the changes that have occurred over time. This can help guide future preservation strategies by identifying which types of buildings or locations are more susceptible to change.

#### 5.4 Limitations

The presented method can be used in countries with advanced geographic base registration systems, like the Netherlands, as the method requires for building topology and corresponding identification numbers to remain consistent over time. However, in countries that do not have a developed registration system, this method is not applicable without further refinements. Additionally, it was not possible to validate all changes as accurately as possible. Preferably one would perform fieldwork validation as this is the most reliable approach to confirm whether detected changes in the data also occurred in reality. By physically verifying changes on-site, one could ensure that false positives (e.g., incorrectly registered buildings) do not distort the analysis. However, field validation is not feasible within the scope of this thesis due to time constraints. Similarly, by employing historical images of the city of Utrecht, which for instance are available in the Utrecht city archives or through Documentatie.org, this validation step can be performed to bridge the gap in earlier time layers.

Furthermore, the object matching method used did not take the shape of buildings into account. Consequently, historical versions of buildings can be matched to new buildings that have an entirely different shape, also causing the subsequent object matching to also be unsuccessful. This is illustrated in Figure 24 below, where a falsely identified merge is shown. To account for this, the BAG ID can be used to validate whether buildings are matched to the same, or to a different building. However, the BGT Pand dataset contains multiple instances where the BAG ID remains unchanged despite the demolition of a building and the construction of a new one in the same location. Molenstraat 16 is an example of this where, according to Google Street View (see Appendix K), a different, now-demolished building stood in 2014, which aligns with the version history in the BGT Pand data. Yet, both the old and new building are assigned the same BAG ID in the BGT Pand dataset. This relates back to the method's dependency on accurate, systemic reporting of changes in the built environment. If this condition is not sufficiently met, as shown in the previously mentioned example, the method's reliability is considerably smaller.

Furthermore, upon validation of the change detection results, it became apparent that building versions marked as demolished were not always demolished in reality. Some buildings were never constructed (*niet gerealiseerd pand*) or incorrectly registered (*pand ten onrechte opgevoerd*) but have still at some point been inputted into the BGT Pand and received a BAG id, despite never having existed in the real world.

Finally, buildings versions can only be attributed a single change type. Therefore, if multiple changes have occurred in a single time step, this cannot be identified using this change detection method. An example of this is shown in Figure 23, where a building was split and partly demolished in a single time step.



Figure 24: Demolition falsely identified as merge

## 6. Conclusion

---

This thesis has sought to study urban transformations in the context of built heritage, using the conservation area of Utrecht as a case study. Its aim was twofold, firstly to research different methods for the vectorization of historical cadastral data at the building level. Secondly, to study the most effective methods for matching and comparing cadastral data from different temporal versions of the BGT. By examining vectorization, object matching and change detection in relation to each other, an attempt was made to create a more holistic approach so that old and new data sources can enrich one another in further research. This study has sought to answer the following research question: *"How can the processes of vectorizing historical cadastral maps and detecting building-level changes over time be combined to analyse the urban transformation of the historical city centre of Utrecht?"*. To help answer the research question, three sub-questions were formulated *"What methods are most effective for the vectorization of historical cadastral data?"*, *"What are the most effective approaches for matching different versions of buildings?"* and *"What methods are most suitable for researching the historical destruction, construction and changes of buildings?"*. Firstly, the sub-questions will be addressed, after which the main question will be answered.

Manual vectorization was found to be an effective, but time-consuming method for the vectorization of historic large scale cadastral maps. Ideally, this approach should be complemented by an automatic vectorization method, which are fast but not sufficiently accurate without model training and parameter adjustment. Here, the vectorized building outlines can serve as training data for a convolutional neural network-based pipeline. By training an automated model using the vectorized data from this thesis, a fundament can be laid enabling researchers to vectorize old map data on a larger scale, holstering the potential of historic data.

A 1:1 object match, using building centroids as the primary matching criterion and supplemented by 'contains' and 'within' spatial relationships, proved to be the most effective method for accurately matching historical buildings to their current counterparts. The method provided some flexibility which was necessary to account for buildings that had undergone slight topology changes over time. This was enriched by linking the BAG to the BGT Pand using the BAG id to verify the matched buildings.

A rule-based approach using seven change classes was found to be most effective for performing change detection. This method was complimented with a refinement step verifying splits and partial demolitions through 1:m spatial joins. This optimization improved the accuracy of detecting building changes in the BGT Pand dataset between 2016 and 2024.

To answer the research question, this thesis demonstrates that combining the processes of vectorizing historical cadastral maps and detecting building-level changes over time enable a more comprehensive analysis of the development of historical city centres, like that of Utrecht. While the vectorization of historical cadastral data created a foundation for linking old and new datasets, the integration of these data sources for change detection proved challenging. Despite difficulties in directly linking the vectorized historical data to modern datasets for change detection, this thesis shows the potential for future work in this area. The methods developed for vectorization and change detection offer a framework that, with further refinement and improved object matching techniques, can lead to a more comprehensive understanding of urban transformation over time. This thesis aims toward bridging the gap between historical and contemporary data, offering a pathway for future research to integrate data for a more holistic analysis of the built environment.

## 7. Recommendations

From the findings of this thesis, several recommendations can be formulated for future research. This thesis has explored the processes of vectorization and object matching and change detection in isolation. In future research, it is encouraged to connect these two interconnected fields. This can be achieved in multiple ways.

Firstly, the vectorization processes presented here have laid a foundation for facilitating a broader base for the extraction of building outlines from historical map data. Future research could use the manually vectorized building data to train automated vectorization methods. This relationship is illustrated by the dashed line between manual and automatic vectorization in Figure 25 below, adapted from the components of the proposed solutions presented in Chapter 3. Additional research in this field would enhance the accessibility of historical data.

Another direction for further study of this topic is to expand the used object matching method to include factors such as building shapes and building pattern constraints. By including these factors, buildings can be matched with more accuracy than through only using an object matching approach. This would enable the method to be employed over a larger time span to research how extracted building data from historical cadastral maps relate to their current forms. Thereby, future studies could expand upon this work by incorporating additional datasets, exploring broader time periods, or comparing different methods for digitizing and analysing historical maps. This is graphically shown by the dashed line between the results and object matching in Figure 25.

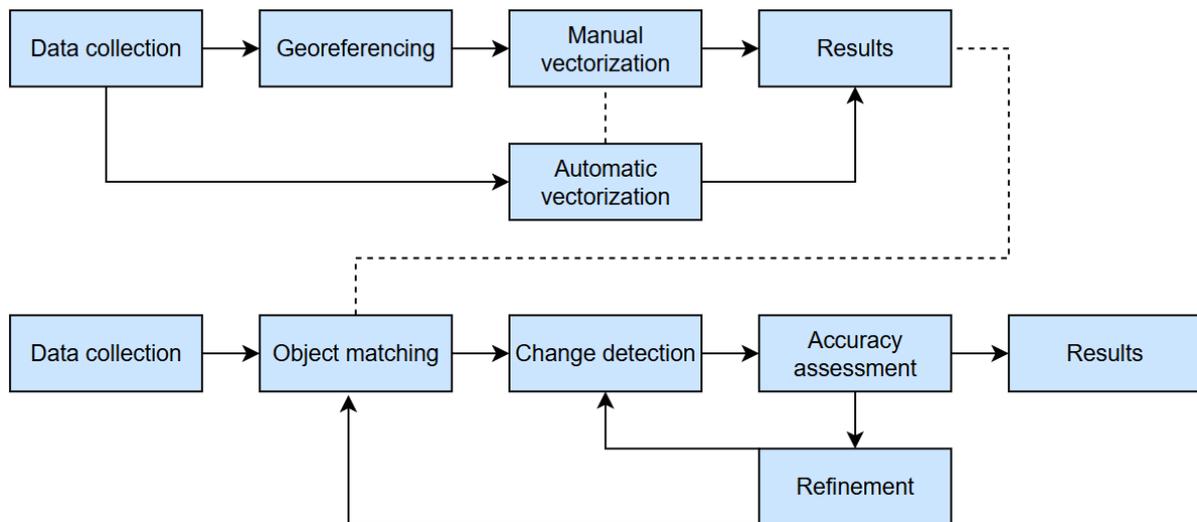


Figure 25: Graphic illustration of recommended research directions

Finally, manual validation was used in this study to validate the results of the object matching and change detection phases. However, since this inherently involves a level of subjectivity, which may introduce inconsistencies when conducting similar research, future research could explore standardized validation methods to reduce human bias in this process.

## Bibliography

- Baiocchi, V., Lelo, K., Milone, M. V., & Mormile, M. (2013). Accuracy of different georeferencing strategies on historical maps of Rome. *Geographia Technica*, No. 1, 2013, pp. 10-16. [https://www.researchgate.net/publication/235988132\\_Accuracy\\_of\\_different\\_georeferencing\\_strategies\\_on\\_historical\\_maps\\_of\\_Rome](https://www.researchgate.net/publication/235988132_Accuracy_of_different_georeferencing_strategies_on_historical_maps_of_Rome)
- Berk, S., & Ferlan, M. (2016). Accurate area determination in the cadaster: case study of Slovenia. *Cartography and Geographic Information Science*, 45(1), 1–17. <https://doi.org/10.1080/15230406.2016.1217789>
- Brovelli, M., & Minghini, M. (2012). Georeferencing old maps: A polynomial-based approach for Como historical cadastres. *E-Perimtron*, 7(3), 97–110. [https://www.e-perimtron.org/Vol\\_7\\_3/Brovelli\\_Minghini.pdf](https://www.e-perimtron.org/Vol_7_3/Brovelli_Minghini.pdf)
- Budig, B., Van Dijk, T. C., Feitsch, F., & Arteaga, M. G. (2016). Polygon consensus: Smart Crowdsourcing for Extracting Building Footprints from Historical Maps. *SIGSPATIAL'16*, 1–4. <https://doi.org/10.1145/2996913.2996951>
- Cascón-Katchadourian, J., & Alberich-Pascual, J. (2021). The Georeferencing of Old Cartography in Geographic Information Systems (GIS): Review, analysis and comparative study of georeferencing software. *Revista General De Información Y Documentación*, 31(1), 437–460. <https://doi.org/10.5209/rgid.76965>
- Chen, N. L., Liao, N. H., Wang, N. J., Fan, N. K., & Hsieh, N. C. (1996). An interpretation system for cadastral maps. *Proceedings of the 13th International Conference of Pattern Recognition*, 3. <https://doi.org/10.1109/icpr.1996.547261>
- Chen, Y., Carlinet, E., Chazalon, J., Mallet, C., Duménieu, B., & Perret, J. (2021a). Vectorization of historical maps using deep edge filtering and closed shape extraction. In *Lecture notes in computer science* (pp. 510–525). [https://doi.org/10.1007/978-3-030-86337-1\\_34](https://doi.org/10.1007/978-3-030-86337-1_34)
- Chen, Y., Carlinet, E., Chazalon, J., Mallet, C., Duménieu, B., & Perret, J. (2021b). Combining deep learning and mathematical morphology for historical map segmentation. In *Lecture notes in computer science* (pp. 79–92). [https://doi.org/10.1007/978-3-030-76657-3\\_5](https://doi.org/10.1007/978-3-030-76657-3_5)
- Chen, Y., Chazalon, J., Carlinet, E., Ngoc, M. Ô. V., Mallet, C., & Perret, J. (2024). Automatic vectorization of historical maps: A benchmark. *PLoS ONE*, 19(2), e0298217. <https://doi.org/10.1371/journal.pone.0298217>
- Dhar, D. B., & Chanda, B. (2006). Extraction and recognition of geographical features from paper maps. *International Journal on Document Analysis and Recognition (IJ DAR)*, 8(4), 232–245. <https://doi.org/10.1007/s10032-005-0010-9>
- Dolfin, M., Kylstra, E.M. & Penders, J. (1989). Utrecht. De huizen binnen de singels. SDU uitgeverij, Den Haag. Rijksdienst voor de Monumentenzorg, Zeist. [https://www.dbnl.org/tekst/dolf001utre01\\_01/index.php](https://www.dbnl.org/tekst/dolf001utre01_01/index.php)
- Drolias, G. C., & Tziokas, N. (2020). Building Footprint Extraction from Historic Maps utilizing Automatic Vectorisation Methods in Open Source GIS Software. *ICA Commission on Cartographic Heritage Into the Digital*, 9–17. <https://doi.org/10.21862/avhm2020.01>

- Esri. (n.d.-a). Using raster cleanup morphological operations. <https://desktop.arcgis.com/en/arcmap/latest/extensions/arcscan/using-raster-cleanup-morphological-operations.htm>
- Esri. (n.d.-b). Overview of Georeferencing—ArcGIS Pro | Documentation. <https://pro.arcgis.com/en/pro-app/latest/help/data/imagery/overview-of-georeferencing.htm#:~:text=The%20first%2Dorder%20polynomial%20transformation,in%20the%20warped%20raster%20dataset.>
- Esri. (n.d.-c). Convolution function. ArcGIS Pro. <https://pro.arcgis.com/en/pro-app/latest/help/analysis/raster-functions/convolution-function.htm>
- Esri. (n.d.-d). Create polygon features. ArcGIS Pro. <https://pro.arcgis.com/en/pro-app/latest/help/editing/create-polygon-features.htm>
- Fan, H., Zipf, A., Fu, Q., & Neis, P. (2014). Quality assessment for building footprints data on OpenStreetMap. *International Journal of Geographical Information Science*, 28(4), 700–719. <https://doi.org/10.1080/13658816.2013.867495>
- Femenia-Ribera, C., Mora-Navarro, G., & Pérez, L. J. S. (2022). Evaluating the use of old cadastral maps. *Land Use Policy*, 114, 105984. <https://doi.org/10.1016/j.landusepol.2022.105984>
- Franken, J. , Florijn, W. , Hoekstra, M. , Hagemans, H.J. (2021). Rebuilding the Cadastral Map of The Netherlands, the Artificial Intelligence Solution. Proceedings of the FIG e-Working Week 21 – 25 June 2021, ‘Smart Surveyors for Land and Water Management - Challenges in a New Reality’, International Federation of Surveyors, Copenhagen, Denmark. ISBN 978-87-92853- 65-3. [https://www.fig.net/resources/proceedings/fig\\_proceedings/fig2021/papers/nl01/NL01\\_jeroen\\_florijn\\_et\\_al\\_11000.pdf](https://www.fig.net/resources/proceedings/fig_proceedings/fig2021/papers/nl01/NL01_jeroen_florijn_et_al_11000.pdf)
- Gemeente Utrecht. (1975). Aanwijzingstekst Beschermd stadsgezicht Binnenstad. [https://cultureelerfgoed.info/Beschermd\\_Gezichten/BG1657/TOELICHTING\\_aanwijzing\\_1657.pdf](https://cultureelerfgoed.info/Beschermd_Gezichten/BG1657/TOELICHTING_aanwijzing_1657.pdf)
- Geonovum. (2020). Basisregistratie Grootchalige Topografie Gegevenscatalogus BGT. Ministerie van Binnenlandse Zaken en Koninkrijksrelaties. <https://docs.geostandaarden.nl/imgeo/catalogus/bgt/#identificatie-en-historie>
- Google. (n.d.). Google Maps Street View. <https://www.google.nl/intl/nl/streetview/>
- Guidotti, R., Monreale, A., Ruggieri, S., Turini, F., Giannotti, F., & Pedreschi, D. (2018). A survey of methods for explaining black box models. *ACM Computing Surveys*, 51(5), 1–42. <https://doi.org/10.1145/3236009>
- Hajiheidari, A. R., Delavar, M. R., & Rajabifard, A. (2023). CADASTRAL AND URBAN MAPS ENRICHMENTS USING SMART SPATIAL DATA FUSION. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, X-4/W1-2022, 263–269. <https://doi.org/10.5194/isprs-annals-x-4-w1-2022-263-2023>

- Heitzler, M., & Hurni, L. (2020). Cartographic reconstruction of building footprints from historical maps: A study on the Swiss Siegfried map. *Transactions in GIS*, 24(2), 442–461. <https://doi.org/10.1111/tgis.12610>
- Herold, H. (2017). Geoinformation from the Past. In *Springer eBooks*. <https://doi.org/10.1007/978-3-658-20570-6>
- Het Utrechts Archief. (n.d.). Utrecht wordt een stad. <https://hetutrechtsarchief.nl/verhalen/40-utrecht-wordt-een-stad>
- Heywood, D. I., Cornelius, S., & Carver, S. (1999). An introduction to geographical information systems. *Choice Reviews Online*, 36(07), 36–3920. <https://doi.org/10.5860/choice.36-3920>
- HisGIS. (n.d.). Over ons. <https://hisgis.nl/over-ons/>
- Huang, Y. (2024). Bibliometric analysis of GIS applications in heritage studies based on Web of Science from 1994 to 2023. *Heritage Science*, 12(1). <https://doi.org/10.1186/s40494-024-01163-y>
- Hulsman, B. (2020). De pleinen van een zieke stad. NRC. <https://www.nrc.nl/nieuws/2020/09/25/de-pleinen-van-een-zieke-stad-a4013281>
- Ignjatić, J., Nikolić, B., Rikalović, A., & Čulibrk, D. (2018). Deep Learning for Historical Cadastral Maps Digitization: Overview, challenges and potential. *Computer Science Research Notes*. <https://doi.org/10.24132/csrn.2018.2803.6>
- Jovanović, D., Gavrilović, M., Sladić, D., Radulović, A., & Govedarica, M. (2021). Building Change Detection Method to support register of identified changes on buildings. *Remote Sensing*, 13(16), 3150. <https://doi.org/10.3390/rs13163150>
- Kadaster. (2024). Basisregistratie Grootschalige Topografie (BGT) - Kadaster.nl zakelijk. <https://www.kadaster.nl/zakelijk/registraties/basisregistraties/bgt>
- Kadaster. (n.d.-a). De geschiedenis van het Kadaster in vogelvlucht. <https://www.kadaster.nl/over-ons/het-kadaster/geschiedenis>
- Kadaster. (n.d.-b). Over BAG - Basisregistratie Adressen en Gebouwen. <https://www.kadaster.nl/zakelijk/registraties/basisregistraties/bag/over-bag>
- Khotanzad, A., & Zink, E. (2003). Contour line and geographic feature extraction from USGS color topographical paper maps. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(1), 18–31. <https://doi.org/10.1109/tpami.2003.1159943>
- Kruizinga P. & Van Doornmalen. S.E.M. (1997). De kadastrale kaart, 1812-1990. Den Haag: Instituut voor Nederlandse Geschiedenis. <https://resources.huygens.knaw.nl/broncommentaren/retro>
- Levi, P. (2009). Digitizing the past, the beginning of a new future: the process of digitizing 12,000 historical maps and making them accessible via the internet. WORLD LIBRARY AND INFORMATION CONGRESS: 75TH IFLA GENERAL CONFERENCE AND COUNCIL. Milan, Italy <http://www.ifla.org/annual-conference/ifla75/index.htm>

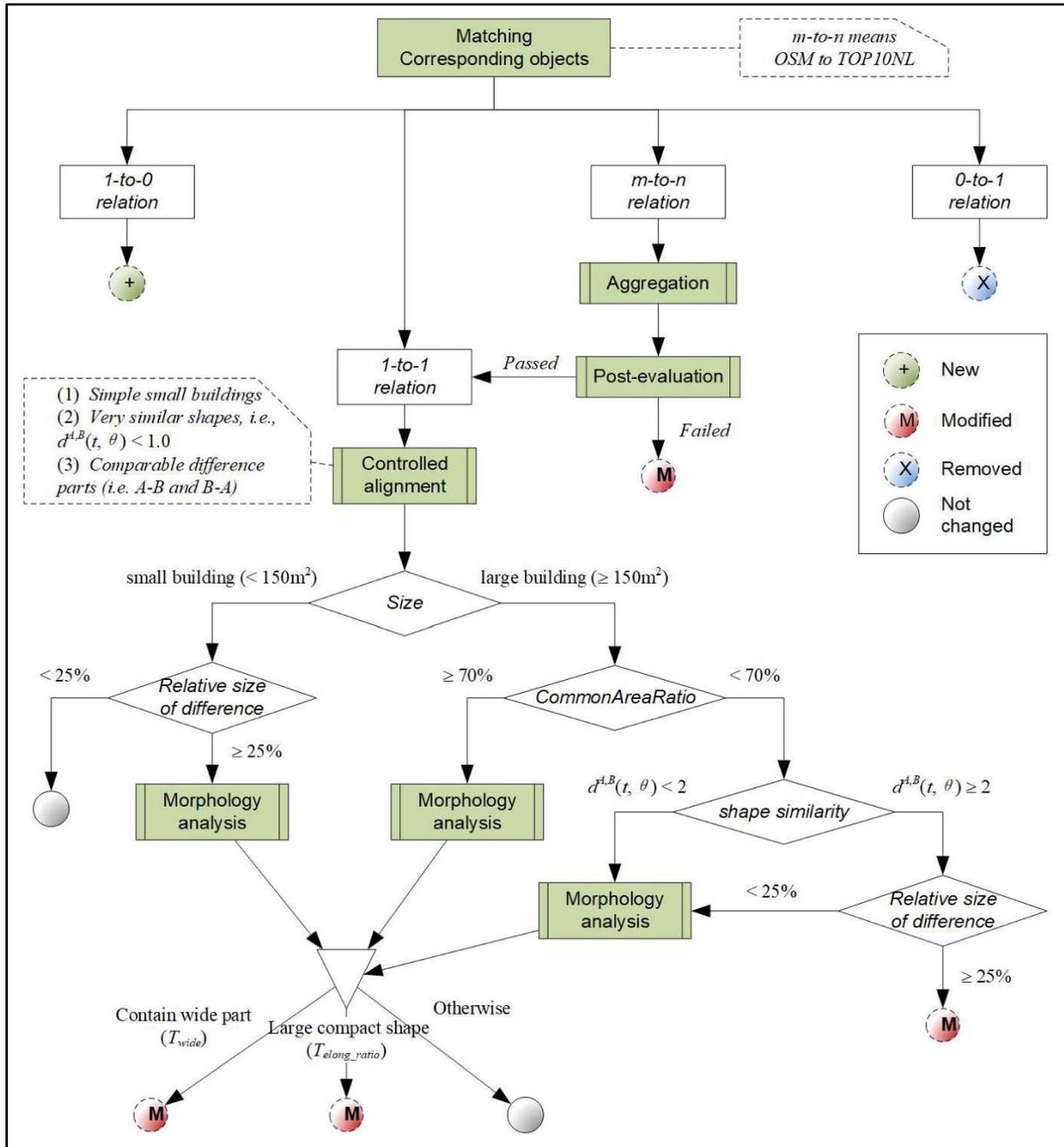
- Liang, W., Ahmad, Y., & Mohidin, H. H. B. (2023). The development of the concept of architectural heritage conservation and its inspiration. *Built Heritage*, 7(1). <https://doi.org/10.1186/s43238-023-00103-2>
- Liu, B., Wu, C., Xu, W., Shen, Y., & Tang, F. (2024). Emerging trends in GIS application on cultural heritage conservation: a review. *Heritage Science*, 12(1). <https://doi.org/10.1186/s40494-024-01265-7>
- Matikainen, L., Hyyppä, J., Ahokas, E., Markelin, L., & Kaartinen, H. (2010). Automatic detection of buildings and changes in buildings for updating of maps. *Remote Sensing*, 2(5), 1217–1248. <https://doi.org/10.3390/rs2051217>
- Monumenten.nl. (2022). Wat is een gemeentelijk monument? <https://www.monumenten.nl/monumenten-overzicht/soorten-monumenten/wat-is-een-gemeentelijk-monument>
- Monumenten.nl. (2024). Soorten monumenten. <https://www.monumenten.nl/soorten-monumenten>
- Navickiene, E., & Riaubiene, E. (2018). Changes of approach to urban context in international guidelines and experiences in Lithuanian urban environment. *Landscape Architecture and Art*, 13, 7–17. <https://doi.org/10.22616/j.landarchart.2018.13.01>
- Oliveira, S. A., Di Leonardo, I. & Kaplan, F. (2017). Machine Vision Algorithms on Cadaster Plans. *Premiere Annual Conference of the International Alliance of Digital Humanities Organizations*. <https://infoscience.epfl.ch/entities/publication/a7d2b5de-3778-4032-bff5-d8435c7c6799>
- PDOK. (n.d.-a). Basisregistratie grootschalige topografie (OGC API). <https://api.pdok.nl/lv/bgt/ogc/v1>
- PDOK. (n.d.-b). Dataset: PDOK Luchtfoto RGB (Open). <https://www.pdok.nl/introductie/-/article/pdok-luchtfoto-rgb-open->
- Petitpierre, R. (2020). Neural networks for semantic segmentation of historical city maps. Cross-cultural performance and the impact of figurative diversity. *Digital Humanities Laboratory (DHLAB)*. <https://arxiv.org/pdf/2101.12478>
- Petitpierre, R. G., Kaplan, F., & Isabella, D. L. (2021). Generic semantic segmentation of historical maps. *CHR 2021: Computational Humanities Research Conference*, Amsterdam, The Netherlands. <https://infoscience.epfl.ch/entities/publication/dfbbd76a-ecca-4c8e-b447-27dbd0d4a46a>
- Petitpierre, R., & Guhenec, P. (2023). Effective annotation for the automatic vectorization of cadastral maps. *Digital Scholarship in the Humanities*, 38(3), 1227–1237. <https://doi.org/10.1093/llc/fqad006>
- Picuno P, Cillis G, Statuto D. (2019). Investigating the time evolution of a rural landscape: How historical maps may provide environmental information when processed using a GIS. *Ecological Engineering*. <https://doi.org/10.1016/j.ecoleng.2019.08.010>
- Povilaitienė, I. (2021). Assessment and Modelling of the Identity of Cityscape: Doctoral dissertation. Technologija: Kaunas, Lithuania. <https://epubl.ktu.edu/object/elaba:100338701/index.html>

- Rensink, E., Theunissen, E. M., & Feiken, H. (2022). Vanuit de lucht zie je meer: remote sensing in de Nederlandse archeologie. Rijksdienst voor het Cultureel Erfgoed. Nederlandse Archeologische Rapporten, 80. ISBN: 978-90-76046-80-8.
- Rijksdienst Voor Het Cultureel Erfgoed. (2024). Beschermde stads- en dorpsgezichten. Ministerie van Onderwijs, Cultuur en Wetenschap.  
<https://www.cultureelerfgoed.nl/onderwerpen/beschermde-stads-en-dorpsgezichten>
- Rijksdienst voor het Cultureel Erfgoed. (n.d.). Beschermde stads- en dorpsgezichten in Nederland.  
[https://kennis.cultureelerfgoed.nl/index.php/Beschermde\\_stads-en\\_dorpsgezichten\\_in\\_Nederland](https://kennis.cultureelerfgoed.nl/index.php/Beschermde_stads-en_dorpsgezichten_in_Nederland)
- Rijksoverheid. (2022). Zorg voor onroerend erfgoed. Erfgoed, Rijksoverheid.nl.  
<https://www.rijksoverheid.nl/onderwerpen/erfgoed/zorg-voor-cultureel-erfgoed/zorg-voor-onroerend-erfgoed>
- Rijksdienst Voor het Cultureel Erfgoed. (2024). Beschermde stads- en dorpsgezichten.  
<https://www.cultureelerfgoed.nl/onderwerpen/beschermde-stads-en-dorpsgezichten>
- Skaloš, J., Weber, M., Lipský, Z., Trpáková, I., Šantrůčková, M., Uhlířová, L., & Kukla, P. (2010). Using old military survey maps and orthophotograph maps to analyse long-term land cover changes – Case study (Czech Republic). *Applied Geography*, 31(2), 426–438.  
<https://doi.org/10.1016/j.apgeog.2010.10.004>
- Southall H, Aucott P, Fleet C, Pert T, Stoner M. (2017). Engaging the public in very large scale gazetteer construction from the ordnance survey “County series” 1: 10,560 mapping of Great Britain. *Journal of Map & Geography Libraries*. 13(1):7–28.  
<https://doi.org/10.1080/15420353.2017.1307305>
- Sze, V., Chen, Y., Yang, T., & Emer, J. (2017). Efficient Processing of deep Neural Networks: A tutorial and survey. *arXiv (Cornell University)*. <https://doi.org/10.48550/arxiv.1703.09039>
- Szypuła, B. (2019). Quality assessment of DEM derived from topographic maps for geomorphometric purposes. *Open Geosciences*, 11(1), 843–865. <https://doi.org/10.1515/geo-2019-0066>
- Thompson, R., & Van Oosterom, P. (2021). Bi-temporal foundation for LADM v2: Fusing event and state based modelling of Land administration data 2D and 3D. *Land Use Policy*, 102, 105246.  
<https://doi.org/10.1016/j.landusepol.2020.105246>
- Tillema, J.A.C. (1975). Bijlage, Schetsen uit de geschiedenis van de monumentenzorg in Nederland. Digitale Bibliotheek voor de Nederlandse Letteren.  
[https://www.dbnl.org/tekst/till014sche01\\_01/till014sche01\\_01\\_0031.php](https://www.dbnl.org/tekst/till014sche01_01/till014sche01_01_0031.php)
- UDS, (n.d.). Documentatie.org. <https://documentatie.org/>
- Utrecht University, (n.d.). Compare maps. Klokan Technologies GmbH.  
<https://uu.oldmapsonline.org/compare#>
- Van Oosterom, P. (1997). Maintaining Consistent Topology including Historical Data in a Large Spatial Database. *Proceedings Auto-Carto 13*, Seattle WA, 1997, 327–336.  
<https://gdmc.nl/oosterom/time.pdf>

- Voulodimos, A., Doulamis, N., Doulamis, A., & Protopapadakis, E. (2018). Deep Learning for Computer Vision: A Brief review. *Computational Intelligence and Neuroscience*, 2018, 1–13. <https://doi.org/10.1155/2018/7068349>
- Yao, Y., Wang, X., Luo, L., Wan, H., & Ren, H. (2023). An Overview of GIS-RS Applications for Archaeological and Cultural Heritage under the DBAR-Heritage Mission. *Remote Sensing*, 15(24), 5766. <https://doi.org/10.3390/rs15245766>
- Zhang, X., Ai, T., Stoter, J., Kraak, M., & Molenaar, M. (2011). Building pattern recognition in topographic data: examples on collinear and curvilinear alignments. *GeoInformatica*, 17(1), 1–33. <https://doi.org/10.1007/s10707-011-0146-3>
- Zhang, X., Stoter, J., Ai, T., Kraak, M., & Molenaar, M. (2013). Automated evaluation of building alignments in generalized maps. *International Journal Of Geographical Information Science*, 27(8), 1550–1571. <https://doi.org/10.1080/13658816.2012.758264>
- Zhang, Y., Jiang, X., & Wang, S. (2022). Fingerspelling Recognition by 12-Layer CNN with Stochastic Pooling. *Mobile Networks and Applications*. <https://doi.org/10.1007/s11036-021-01900-8>
- Zhou, X., Chen, Z., Zhang, X., & Ai, T. (2018). Change Detection for Building Footprints with Different Levels of Detail Using Combined Shape and Pattern Analysis. *ISPRS International Journal of Geo-Information*, 7(10), 406. <https://doi.org/10.3390/ijgi7100406>

# Appendices

## Appendix A: The rule system for change detection. Zhou et al., 2018.



The outline of the rule system for change detection. Zhou et al., 2018.

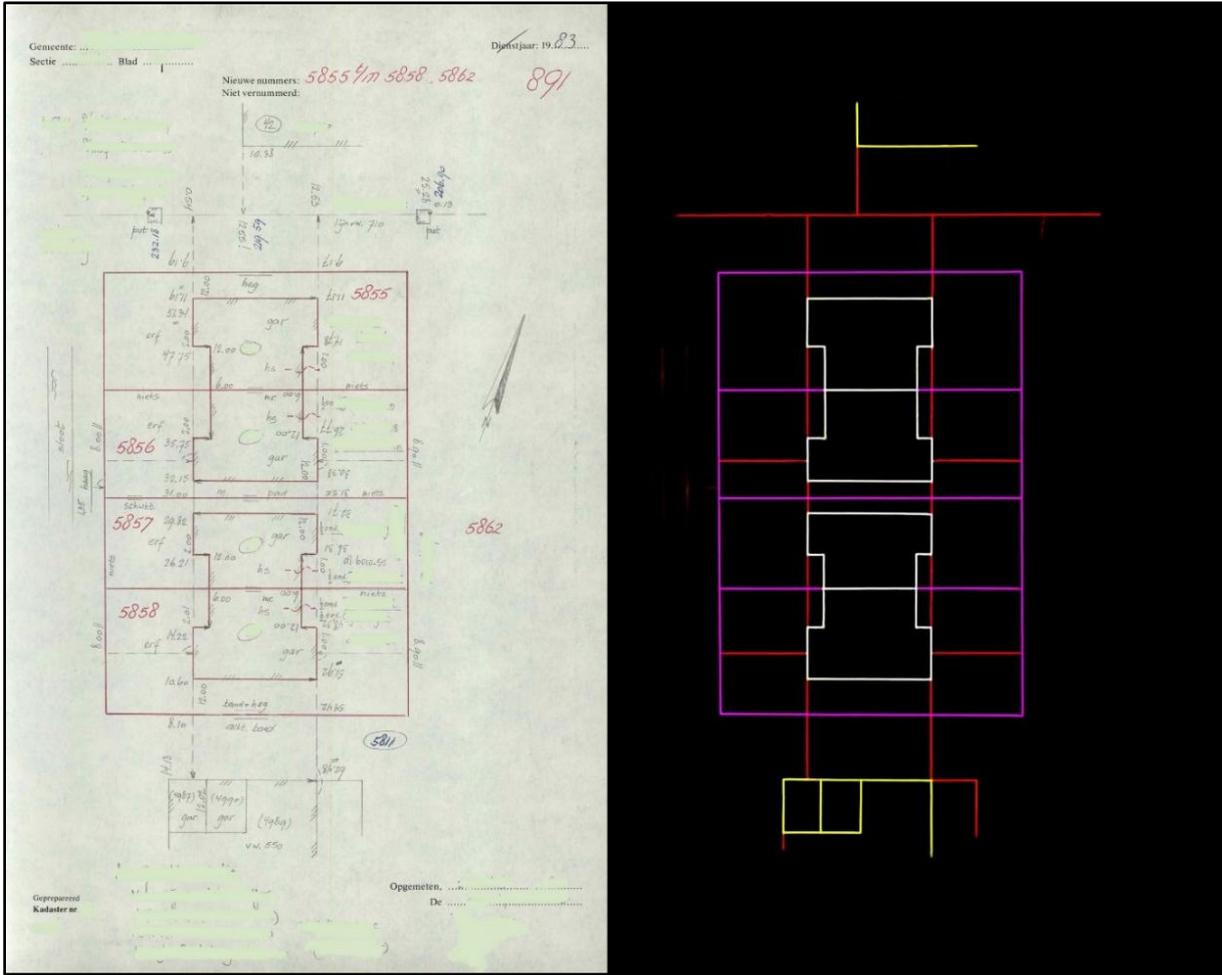
Appendix B: Map of Utrecht in the twelfth century. Van Der Vlerk, 1983.



Appendix C: 1965 large scale cadastral map. The National Cadastre, 1965.



Appendix D: Field sketch and VeCTOR model output. Franken et al., 2021.



## Appendix E: Code for object matching I

```
import arcpy

arcpy.env.workspace = r"C:\Users\boris\OneDrive - Universiteit
Utrecht\ArcGIS\Projects\ChangeDetection\ChangeDetection.gdb"

# Define input feature classes
old_buildings = "OudePandVersies"
new_buildings = "BGThuidig"

# Output feature class
output_layer = r"C:\Users\boris\OneDrive - Universiteit
Utrecht\ArcGIS\Projects\ChangeDetection\ChangeDetection.gdb\zAreaComparisonResult"

# Step 1: Perform a Spatial Join to find intersecting features
spatial_join_output = "zspatial_join_output"
arcpy.analysis.SpatialJoin(
    target_features=old_buildings,
    join_features=new_buildings,
    out_feature_class=spatial_join_output,
    join_type="KEEP_COMMON",
    match_option="HAVE_THEIR_CENTER_IN"
)

# Step 2: Add a field to calculate Area Difference Ratio
arcpy.management.AddField(spatial_join_output, "Area_Diff_Ratio", "DOUBLE")

# Step 3: Calculate Area Difference Ratio
with arcpy.da.UpdateCursor(spatial_join_output,
    ["geom_Area", "geom_Area_1", "Area_Diff_Ratio"]) as cursor:
    for row in cursor:
        old_area = row[0]
        new_area = row[1]
        if old_area > 0:
            row[2] = abs(new_area - old_area) / old_area
        else:
            row[2] = None
        cursor.updateRow(row)

# Step 4: Select features where the Area Difference Ratio <= 0.1
arcpy.management.SelectLayerByAttribute(
    in_layer_or_view=spatial_join_output,
    selection_type="NEW_SELECTION",
    where_clause="Area_Diff_Ratio <= 0.1"
)

# Step 5: Export selected features to a new feature class
arcpy.management.CopyFeatures(spatial_join_output, output_layer)

# Step 6: Add the output to the active map
aprx = arcpy.mp.ArcGISProject("CURRENT")
map_view = aprx.activeMap
map_view.addDataFromPath(output_layer)
```

## Appendix F: Code for object matching II

```
import arcpy

arcpy.env.workspace = r"C:\Users\boris\OneDrive - Universiteit
Utrecht\ArcGIS\Projects\ChangeDetection\ChangeDetection.gdb"

# Define input feature classes
old_buildings = "OudePandVersies"
new_buildings = "OudePandVersiesOngematched"

# Output feature class
output_layer = r"C:\Users\boris\OneDrive - Universiteit
Utrecht\ArcGIS\Projects\ChangeDetection\ChangeDetection.gdb\zAreaComparisonResultOPVO"

# Step 1: Perform a Spatial Join to find intersecting features
spatial_join_output = "zspatial_join_outputOPVO"
arcpy.analysis.SpatialJoin(
    target_features=old_buildings,
    join_features=new_buildings,
    out_feature_class=spatial_join_output,
    join_type="KEEP_COMMON",
    match_option="CONTAINS"
)

# Step 2: Add a field to calculate Area Difference Ratio
arcpy.management.AddField(spatial_join_output, "Area_Diff_Ratio", "DOUBLE")

# Step 3: Calculate Area Difference Ratio
with arcpy.da.UpdateCursor(spatial_join_output,
    ["geom_Area", "geom_Area_1", "Area_Diff_Ratio"]) as cursor:
    for row in cursor:
        old_area = row[0]
        new_area = row[1]
        if old_area > 0:
            row[2] = abs(new_area - old_area) / old_area
        else:
            row[2] = None
        cursor.updateRow(row)

# Step 4: Select features where the Area Difference Ratio <= 0.1
arcpy.management.SelectLayerByAttribute(
    in_layer_or_view=spatial_join_output,
    selection_type="NEW_SELECTION",
    where_clause="Area_Diff_Ratio <= 0.1"
)

# Step 5: Export selected features to a new feature class
arcpy.management.CopyFeatures(spatial_join_output, output_layer)

# Step 6: Add the output to the active map
aprx = arcpy.mp.ArcGISProject("CURRENT")
map_view = aprx.activeMap
map_view.addDataFromPath(output_layer)
```

## Appendix G: Code for change detection

```
arcpy.env.workspace = r"C:\Users\boris\OneDrive - Universiteit
Utrecht\ArcGIS\Projects\ChangeDetectionSplitMerge.gdb"
adr10 = "ADR10"

# Output layers
expanded_buildings = "ExpandedBuildings"
merged_buildings = "MergedBuildings"
split_buildings = "SplitBuildings"
partly_demolished = "PartlyDemolished"

# Step 1: Add the ChangeType field if it doesn't already exist
arcpy.management.AddField(adr10, "ChangeType", "TEXT")

# Step 2: Classify changes based on ADR values
with arcpy.da.UpdateCursor(adr10, ["ADR", "Shape_Area", "geom_Area_1", "ChangeType"]) as cursor:
    for row in cursor:
        adr_value = row[0]

        print(f"ADR: {adr_value}")

        # Classify based on ADR values
        if adr_value is not None:
            if adr_value > 0.1 and adr_value <= 0.6:
                row[3] = "Expanded"
                print("Classified as Expanded")
            elif adr_value < -0.3:
                row[3] = "Split"
                print("Classified as Split")
            elif adr_value > 0.6:
                row[3] = "Merged"
                print("Classified as Merged")
            elif -0.3 <= adr_value <= -0.1:
                row[3] = "Partly Demolished"
                print("Classified as Partly Demolished")

        cursor.updateRow(row)

# Step 3: Export categorized layers
arcpy.management.SelectLayerByAttribute(
    in_layer_or_view=adr10,
    selection_type="NEW_SELECTION",
    where_clause="ChangeType = 'Expanded'"
)
arcpy.management.CopyFeatures(adr10, expanded_buildings)

arcpy.management.SelectLayerByAttribute(
    in_layer_or_view=adr10,
    selection_type="NEW_SELECTION",
    where_clause="ChangeType = 'Merged'"
)
arcpy.management.CopyFeatures(adr10, merged_buildings)

arcpy.management.SelectLayerByAttribute(
    in_layer_or_view=adr10,
```

```
    selection_type="NEW_SELECTION",
    where_clause="ChangeType = 'Split'"
)
arcpy.management.CopyFeatures(adr10, split_buildings)

arcpy.management.SelectLayerByAttribute(
    in_layer_or_view=adr10,
    selection_type="NEW_SELECTION",
    where_clause="ChangeType = 'Partly Demolished'"
)
arcpy.management.CopyFeatures(adr10, partly_demolished)
```

## Appendix H: Code refinement step I

```
import arcpy

arcpy.env.workspace = r"C:\Users\boris\OneDrive - Universiteit Utrecht\ArcGIS\Projects\SplitTestBAGid.gdb"
arcpy.env.overwriteOutput = True # Allow overwriting existing outputs

# Define layers
BGT_old = "SplitBAGtestRDnew" # Old BGT buildings
BGT_new = "BGTutrechtGOED" # New BAG buildings

# Create feature layers
arcpy.MakeFeatureLayer_management(BGT_old, "BGT_Layer")
arcpy.MakeFeatureLayer_management(BGT_new, "BAG_Layer")

# Select BAG buildings within BGT buildings
arcpy.SelectLayerByLocation_management("BAG_Layer", "WITHIN", "BGT_Layer")

# Count selected features
count = int(arcpy.GetCount_management("BAG_Layer")[0])
print(f"Number of matched BAG buildings: {count}")

if count > 0:
    # Perform Spatial Join directly without exporting extra layers
    joined_bag = "BAG_with_BGT_Join"
    arcpy.SpatialJoin_analysis("BAG_Layer", "BGT_Layer", joined_bag,
                               "JOIN_ONE_TO_MANY", match_option="WITHIN")
    print(f"Spatial Join complete. Output: {joined_bag}")

    # Generate summary table to count how many BAG buildings per BGT building
    summary_table = "BGT_Split_Stats"
    arcpy.Statistics_analysis(joined_bag, summary_table, [{"main__BGTutrecht_bag_pnd", "COUNT"}],
                              "main__BGTutrecht_bag_pnd")
    print(f"Summary table created: {summary_table}")

    # Check field names
    fields = [f.name for f in arcpy.ListFields(summary_table)]
    print("Fields in BGT_Split_Stats:", fields)

    # Use the correct field name for COUNT (ArcGIS might rename it)
    count_field = "COUNT_" if "COUNT_" in fields else "FREQUENCY" if "FREQUENCY" in fields else "COUNT"

    # Select only BAG buildings that occur more than once (actual splits)
    arcpy.MakeFeatureLayer_management(joined_bag, "Joined_Layer")
    arcpy.AddJoin_management("Joined_Layer", "main__BGTutrecht_bag_pnd", summary_table,
                              "main__BGTutrecht_bag_pnd")

    # Select only rows where COUNT > 1
    selection_query = f"{summary_table}.{count_field} > 1"
    arcpy.SelectLayerByAttribute_management("Joined_Layer", "NEW_SELECTION", selection_query)

    # Specify a new unique output name
    final_output = "BAG_with_BGT_Join_Split_Buildings"

    # Copy the selected features to the new output dataset
    arcpy.CopyFeatures_management("Joined_Layer", final_output)
    print(f"Filtered only split BAG buildings. Updated: {final_output}")
```

## Appendix I: Code refinement step II

```
import arcpy

arcpy.env.workspace = r"C:\Users\boris\OneDrive - Universiteit Utrecht\ArcGIS\Projects\SplitTestBAGid.gdb"

# Define layers
PartlyDemolished = "PartlyDemBAGtest" # Partly demolished buildings
BGTutrechtGOED = "BGTutrechtGOED" # Updated new BAG buildings

# Perform a spatial selection using WITHIN to select BAG buildings inside Partly Demolished buildings
arcpy.MakeFeatureLayer_management(PartlyDemolished, "Demolished_Layer")
arcpy.MakeFeatureLayer_management(BGTutrechtGOED, "BGT_Layer")

# Select new BAG buildings within partly demolished buildings
arcpy.SelectLayerByLocation_management("BGT_Layer", "HAVE_THEIR_CENTRE_IN", "Demolished_Layer")

# Perform a spatial join to link Partly Demolished buildings to the selected new BAG buildings
demolished_bgt_bag_join = "Demolished_BGT_BAG_Join"
arcpy.analysis.SpatialJoin("Demolished_Layer", "BGT_Layer", demolished_bgt_bag_join,
join_type="KEEP_COMMON")

# Summarize the count of new BAG buildings for each Partly Demolished building
summary_table = "Summary_PartlyDemolished_BGT_BAG"
arcpy.analysis.Statistics(demolished_bgt_bag_join, summary_table, [{"Join_Count", "COUNT"}],
"main__BGTutrecht_bag_pnd")

# Make a table view from the summary table
arcpy.MakeTableView_management(summary_table, "Summary_View")

# Select buildings with more than one new BAG building associated (indicating splits)
arcpy.SelectLayerByAttribute_management("Summary_View", "NEW_SELECTION", "COUNT > 1")

# Copy the filtered result into a new layer (split buildings)
split_buildings_filter = "Split_PartlyDemolished_Buildings"
arcpy.TableToTable_conversion("Summary_View", arcpy.env.workspace, split_buildings_filter)

# Select Split features in Partly Demolished class
selected_values = [
'344100000007254', '344100000011132', '344100000073288', '344100000002564',
'344100000012243', '344100000024396', '344100000025433', '344100000029794',
'344100000029848', '344100000041667', '344100000049912', '344100000051095',
'344100000051098', '344100000058748', '344100000079460'
]

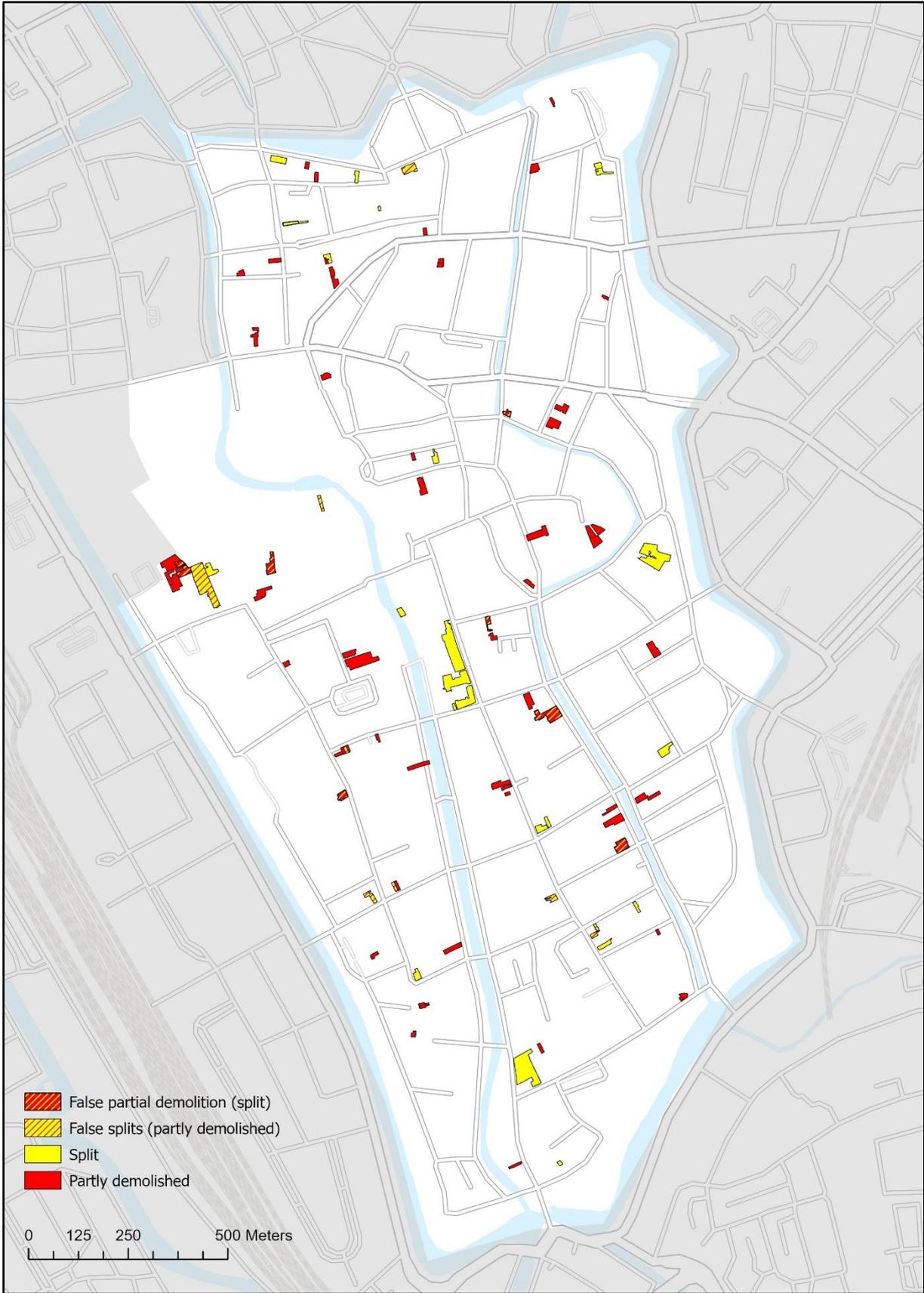
# Strip any leading/trailing spaces from the selected values
selected_values = [val.strip() for val in selected_values]

# Construct the query string, ensuring values are properly quoted
query = "main__BGTutrecht_bag_pnd IN ('" + "'".join(selected_values) + "')"

# Check if the query is being constructed properly
print(f"Query: {query}")

# Select the features from PartlyDemBAGtest based on the query
arcpy.SelectLayerByAttribute_management("PartlyDemBAGtest", "NEW_SELECTION", query)
```

Appendix J: Output refinement step



Appendix K: Demolished- and newly constructed buildings. Google, n.d.

