Connecting digital and physical representations through semantics and geometry

GUSTAF UGGLA

Licentiate Thesis
Stockholm, Sweden 2019

Avdelningen för Geodesi och satellitpositionering
Institutionen för Fastigheter och byggande
Kungliga Tekniska högskolan
100 44 Stockholm

TRITA-ABE-DLT-1914
ISBN 978-91-7873-196-1

Akademisk avhandling som med tillstånd av Kungl Tekniska högskolan framlägges till offentlig granskning för avläggande av licentiatexamen torsdagen 23 maj 2019 klockan 10.00 i sal V3, Kungl Tekniska högskolan, Teknikringen 72, Stockholm.

Tryck: Universitetsservice US AB

Abstract

The fields of geodesy and building information modeling (BIM) meet each other in the intersection between the physical and the digital world. Within the construction industry, the role of geodesy has typically been to describe the position of assets and to transform the geometries of those assets between coordinate systems suitable for design and coordinate systems with a known relation to the Earth. This is not changed by the introduction of BIM but rather emphasized by it, as higher degrees of automation and prefabrication increases the need for strict and non-distorting transformations. The object-oriented aspects of BIM require that captured geodata can be semantically classified and that objects can be reconstructed and extracted from the geodata. In this landscape, geodesy is the bridge between model and reality, connecting the two worlds through both semantics and geometry. This thesis is a comprehensive summary of three papers within these two topics. The first paper describes the geometric transformations required throughout the life cycle of a built asset and assesses the georeferencing capabilities of the open BIM standard Industry Foundation Classes (IFC). The second and third paper propose and showcase a methodology where image-based deep learning is used to extract roadside objects from mobile mapping data. The findings of the first paper include suggestions for how IFC can be improved in order to facilitate better georeferencing, and the second and third paper show that the proposed methodology performs well in comparison to a manual classification.

Sammanfattning

De två områdena geodesi och byggnadsinformationsmodellering (BIM) möter varandra i skärningspunkten mellan den fysiska och den digitala världen. Inom byggindustrin har geodesins roll historiskt varit att positionsbestämma anläggningar samt att transformera deras geometrier mellan koordinatsystem lämpliga antingen för design eller för inmätning och utsättning. Detta har inte ändrats av att BIM börjat användas, utan det har snarare blivit ännu viktigare då högre nivåer av automatisering och prefabricering ställer högre krav på strikta och icke-deformerande transformationer. De objektorienterade aspekterna av BIM kräver att infångade geodata kan klassificeras semantiskt och att objekt kan återskapas och extraheras från dessa geodata. I detta landskap utgör geodesin en bro mellan modell och verklighet, och sammanlänkar dessa världar genom både semantik och geometri. Denna avhandling är en sammanfattning av tre artiklar inom dessa två områden. Den första artikeln beskriver de geometriska transformationer som krävs genom en anläggnings livscykel och utvärderar georefereringsförmågan hos den öppna BIM-standarden Industry Foundation Classes (IFC). Den andra och tredje artikeln föreslår och demonstrerar en metod där bildbaserad deep learning används för att extrahera vägnära objekt ur data insamlat genom mobile mapping. Slutsatserna från den första artikeln inkluderar förslag på hur IFC kan utvecklas för att möjliggöra bättre georeferering, och de två andra artiklarna visar att den föreslagna metoden presterar väl i jämförelse med en manuell klassificering.

Acknowledgments

I would like to thank my main supervisor Milan Horemuz for his never ending support, guidance, and for keeping a door that is always open. I would also like to thank my co-supervisors Väino Tarandi and Patric Jansson for their support and guidance, as well as Ingemar Lewén, Johan Vium Andersson, and Magnus Larson, whose expertise and input has been of great value to this work. Finally I would like to thank Mohammad Bagherbandi for his review of this thesis.

This work is funded by Trafikverket (Swedish Transport Administration), grant FUD 6240.

Contents

List of Papers

Paper 1 Uggla, G. and Horemuz, M. (2018). Geographic capabilities and limitations of Industry Foundation Classes. *Automation in Construction*, 96:554–566

Paper 2 Uggla, G. (2019). Classification and object reconstruction in point clouds using semantic segmentation and transfer learning. In *Proceedings of the International Council for Research and Innovation in Building and Construction World Building Congress 2019*, Hong Kong. Accepted.

Paper 3 Uggla, G. (2019). Automatic extraction of roadside objects from mobile mapping data. Manuscript submitted for publication.

List of Figures

Acronyms

AEC Architecture, Engineering, and Construction

AMG Automated Machine Guidance

BIM Building Information Modeling

CNN Convolutional Neural Network

CRS (geodetic) Coordinate Reference System

FCN Fully Convolutional Network

GNSS Global Navigation Satellite System

GPS Global Positioning System

IFC Industry Foundation Classes

LIDAR Light Detection and Ranging

MLP Multi-Layer Perceptron

MMS Mobile Mapping System

PPM Parts Per Million

UTM Universal Transverse Mercator

Chapter 1

Introduction

The adoption of building information modeling (BIM) is thoroughly changing the way information is managed within the architecture, engineering, and construction (AEC) industry and has been described as a shift of paradigm (Azhar, 2011). The descriptions and definitions of BIM are somewhat varied, but they all share certain key characteristics. BIM aims to replace the document based communication used in the AEC industry with an object-oriented data model containing all information necessary throughout the life cycle of a built asset (Eastman et al., 2011).

The scope of BIM ranges from a more advanced form of 3D computer aided design (CAD) (Yan and Demian, 2008) to more holistic approaches that include processes, technologies, and policies (Succar, 2009). Future visions of BIM include synchronization of the model with real time data and the concept of the digital twin[1] (Delbrügger et al., 2017). The benefits commonly attributed to BIM include improvements to the quality of the final product, improved collaboration between actors, and improved efficiency and sustainability (Azhar, 2011; Eastman et al., 2011).

The fields of geodesy and BIM meet each other at the intersection between the physical and the digital world. Geodesy acts as a bridge that can transform a physical object into a digital representation and vice versa. This transformation includes both the geometry of an object as well as its semantics. Not all data in BIM is geometric, but all geometric data in BIM does, either directly or indirectly, have a geographic location. Objects created in the design phase of a construction project are supposed to be constructed in a specific location and with a specific orientation, and they are commonly designed in relation to existing geodata. BIM models created for existing assets have a direct geographic location the same way the physical asset has one. A digitally created building module aimed for mass-

[1] "The vision of the Digital Twin itself refers to a comprehensive physical and functional description of a component, product or system, which includes more or less all information which could be useful in all — the current and subsequent — lifecycle phases"(Boschert and Rosen, 2016)

production might not have a direct geographic location associated with it, but must still at some point be georeferenced in order to be constructed.

Data is moving from the terrain to geodetic coordinate reference systems to local Cartesian coordinate systems and back throughout the life cycle of a built asset. Data regarding the terrain is captured in a local Cartesian coordinate system, georeferenced in a geodetic reference frame, and rendered in a local Cartesian coordinate system where new data can also be created. The newly created data is then transformed to a geodetic reference frame and, finally, constructed in the terrain. The creation of an as-built model then requires that the physical asset is surveyed and georeferenced, and any alteration made during the operational phase of the asset should result in changes being made to both the physical and digital version, which in turn might require more surveying and more transformations. Paper 1 includes a general description of the transformations between the terrain, geodetic reference frames, and local Cartesian coordinate systems, as well as an assessment of the georeferencing capabilities of the open BIM standard Industry Foundation Classes (IFC).

The semantic and object-oriented nature of BIM affects the acquisition of geodata, especially for tasks such as the creation of as-built models, where the captured geometric information must be segmented into discrete objects with clear semantic definitions. This puts an emphasis on the automatic processing of data such as point clouds, where an unstructured list of points has to be transformed into objects with parametric geometries and semantic classifications. The available processing methods should be a natural factor regarding the choice of method for data acquisition.

The rapid development in the field of machine learning and the use of deep neural networks, which is also known as deep learning, has drastically changed the landscape regarding what can and cannot be understood by a machine (LeCun et al., 2015). Computers are now able to interpret images, sound, and language in an unprecedented way and have even surpassed human performance in the task of image recognition (Russakovsky et al., 2015; He et al., 2016). Possible applications for image-based classification and object identification are investigated in Papers 2 and 3. There are many more possibilities to explore within the use of deep learning for geodata processing, such as the network architectures PointNet and PointNet++ developed by Qi et al. (2017a) and Qi et al. (2017b), and there are likely many more to come, considering the efforts and resources spent in the field of machine learning.

1.1 Objective and contribution

This thesis investigates the role of geodesy in relation to BIM, where the handling of geodetic information in BIM throughout the life cycle of a built asset and the identification of semantics in geodata have been identified as two main research topics. The first topic covers the coordinate systems, transformations, and geodetic

metadata that are required to work with BIM, and the second topic covers how objects can be extracted and reconstructed from unstructured geodata such as point clouds.

The objective of this thesis is to:

- Assess and evaluate the georeferencing capabilities of IFC
- Investigate the potential of using image segmentation to extract objects from point clouds

There is a strong focus on automation throughout this thesis. The aim is to develop and enable systems that can act autonomously and with minimal requirements for human intervention. In the case of IFC, there should be no room for ambiguity regarding the coordinate or reference system the geometries are stored in, and the transformations needed to construct the geometries on the surface of the Earth should be clearly defined. Any prefabricated elements should have the correct dimensions for the actual site, and automated machine guidance (AMG) systems should be able to use the data in the IFC file without any additional information. The aim for the object identification presented in this thesis is to create fully automatic pipelines that directly can consume the data outputted from the capturing system and extract semantic information and geometries for various objects in the scene. The only manual work required should be the creation of training data.

The contribution of this thesis consists of, apart from the recognition of the interaction between the fields of BIM and geodesy, an assessment of the current status regarding the georeferencing of IFC models, suggestions for improvements to IFC to facilitate better georeferencing, and a methodology with example implementations showing how the outstanding performance of state-of-the-art image segmentation technology can be used to semantically classify and extract objects from point clouds.

1.2 Thesis structure

This thesis is written as a comprehensive summary including three papers:

Paper 1 Uggla, G. and Horemuz, M. (2018). Geographic capabilities and limitations of Industry Foundation Classes. *Automation in Construction*, 96:554–566

Paper 2 Uggla, G. (2019). Classification and object reconstruction in point clouds using semantic segmentation and transfer learning. In *Proceedings of the International Council for Research and Innovation in Building and Construction World Building Congress 2019*, Hong Kong. Accepted.

Paper 3 Uggla, G. (2019). Automatic extraction of roadside objects from mobile mapping data. Manuscript submitted for publication.

The structure of this thesis is as follows: Chapter 1 introduces the concept of BIM, the relationship between BIM and geodesy, and describes the objective of the thesis. Chapter 2 describes the principles of geometrical geodesy used in Paper 1 as well as the results from Paper 1, Chapter 3 gives a conceptual description of deep learning, describes the general methodology used in Papers 2 and 3, and presents the results from Papers 2 and 3, and Chapter 4 presents the conclusions of the thesis and suggests directions for future research within the identified topics.

1.3 Declaration of contributions

Paper 1 Gustaf Uggla wrote the paper, implemented the proposed georeferencing methods, and performed all computations. Milan Horemuz provided supervision and feedback as well as expertise regarding the theoretical geodesy. The research objective and the proposed georeferencing methods were developed as a collaboration between the two authors.

Paper 2 Gustaf Uggla is the sole author of the paper.

Paper 3 Gustaf Uggla is the sole author of the paper.

Chapter 2

Georeferencing local data to a geodetic coordinate reference system

A central problem in the BIM workflow is the transferring of geometries from the digital representation to the physical world and back. Paper 1 assesses the possibilities offered by IFC in this regard, identifies certain weaknesses in the current standard, and suggests improvements to IFC that would make the standard compatible with better georeferencing practices.

2.1 Coordinate systems

In this thesis, a coordinate system is a system where a position can be described using numbers, and there are several different coordinate systems used throughout the life cycle of a built asset. A built asset or its digital representation can exist in the physical world, which is also referred to as the terrain, in a geodetic coordinate reference system (CRS), which is a coordinate system used to describe positions relative to the Earth, in a map projection, which is a planar approximation of a CRS, or in a local Cartesian coordinate system, which is a Euclidean coordinate system with orthogonal axes that is commonly used in design software.

CRSs are described in Section 2.1.1, map projections in Section 2.1.2, and local Cartesian coordinate systems in Section 2.1.3. Apart from these coordinate systems that are used for positioning there are also height systems, which are briefly explained in Section 2.1.4.

2.1.1 Geodetic coordinate reference systems

A CRS consists of a reference ellipsoid and a datum point, which ties the ellipsoid to the physical Earth. The ellipsoid is an ellipsoid of revolution and it is defined

by its semi-major and semi-minor axes, a and b, or by its semi-major axis and the flattening f:

$$f = \frac{a - b}{a} \tag{2.1}$$

Geodetic coordinates for a point P are described in terms of latitude (ϕ), which is the angle between the equatorial plane and a line that is perpendicular to the surface of the ellipsoid and passing through P, longitude (λ), which is the angle between the Greenwich meridian plane and the local meridian plane at P, and ellipsoidal height (h), which is the distance between the surface of the ellipsoid and P along a line perpendicular to the surface of the ellipsoid, see Figure 2.1.

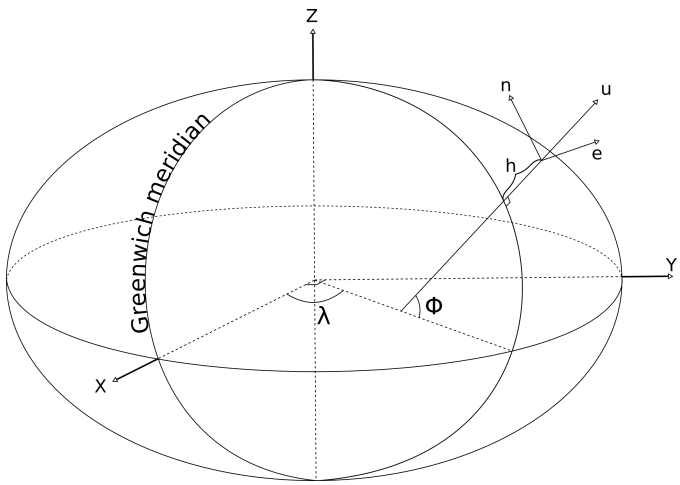

Figure 2.1: A reference ellipsoid with the definition of the Greenwich meridian, latitude ϕ, longitude λ, height over ellipsoid h, the Earth Centered Earth Fixed (ECEF) coordinate system (X, Y, Z), and the oriented engineering system (e, n, u)

Another coordinate system used to describe position relative to the reference ellipsoid is the Earth centered Earth fixed (ECEF) coordinate system (X, Y, Z). X is a vector in the intersection between the equatorial plane and the Greenwich meridian plane, oriented towards the Greenwich meridian, Y is a vector in the equatorial plane perpendicular to X, and Z is perpendicular to both X and Y, forming a right-handed coordinate system, see Figure 2.1.

Transformation from geodetic coordinates to ECEF coordinates is possible using (Hofmann-Wellenhof et al., 2008):

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} (\eta + h)\, cos\phi\, cos\lambda \\ (\eta + h)\, cos\phi\, sin\lambda \\ [\eta(1 - e^2) + h]\, sin\phi \end{bmatrix} \tag{2.2}$$

where e is the first eccentricity of the ellipsoid (Hofmann-Wellenhof et al., 2008):

$$e^2 = 2f - f^2 \tag{2.3}$$

and where η is the radius of curvature in the prime vertical at ϕ (Hofmann-Wellenhof et al., 2008):

$$\eta = \frac{a}{\sqrt{1 - e^2 sin^2\phi}} \tag{2.4}$$

The longitude λ can be calculated exactly from ECEF coordinates using (Hofmann-Wellenhof et al., 2008):

$$tan(\lambda) = \frac{Y}{X} \tag{2.5}$$

and latitude ϕ and ellipsoidal height h can be calculated iteratively by using (Hofmann-Wellenhof et al., 2008):

$$\begin{aligned} tan(\phi_n) &= \frac{Z}{\sqrt{X^2+Y^2}(1-e^2\frac{\eta_{n-1}}{\eta_{n-1}+h_{n-1}})} \\ \eta_n &= \frac{a}{\sqrt{1-e^2 sin^2(\phi_n)}} \\ h_n &= \frac{\sqrt{X^2+Y^2}}{cos(\phi_n)} - \eta_n \end{aligned} \tag{2.6}$$

where a is the semi-major axis of the reference ellipsoid and where n is the current iteration.

2.1.2 Map projections

A map projection is a mathematical projection of latitude and longitude to a planar surface. Coordinates in a map projection are typically denominated as northing (N) and easting (E), and are functions of latitude, longitude, and the axes of the ellipsoid:

$$N = f_N(\phi, \lambda, a, b) \qquad E = f_E(\phi, \lambda, a, b) \tag{2.7}$$

There are three main types of map projections; azimuthal projections, conical projections, and cylindrical projections, see Figure 2.2.

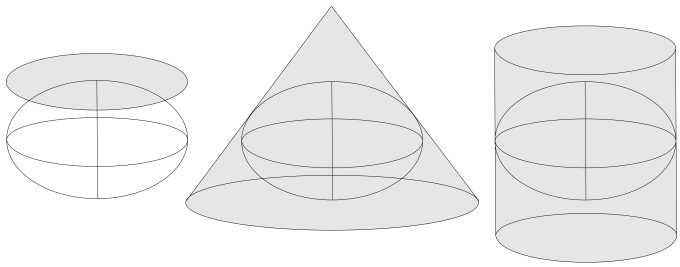

Figure 2.2: From left to right: azimuthal projection, conical projection, and cylindrical projection

For azimuthal projections, the projection surface is a plane, for conical projections, the projection surface is a cone, and for cylindrical projections, the projection surface is a cylinder. The projection surface can either be tangent to the reference ellipsoid, as in Figure 2.2, or it can be secant, which means that the surface intersects the reference ellipsoid.

It is impossible to project a curved surface to a plane without distorting distances, angles, or areas, and map projections are often categorized according to the distortions they produce. Equidistant map projections maintain distances in certain directions, conformal map projections maintain angles, and equal-area projections maintain areas. The transverse Mercator projection, see Figure 2.3, is a commonly used cylindrical and conformal map projection that is well suited to render areas extended along the North-South axis.

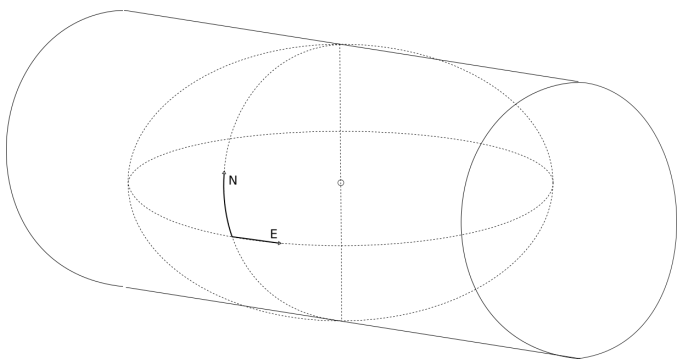

Figure 2.3: Transverse Mercator projection

For a transverse Mercator that is tangent to the reference ellipsoid, as in Figure 2.3, any distance along its central meridian that is on the surface of the ellipsoid will not be distorted when projected to the map plane. For a transverse Mercator that is secant to the reference ellipsoid, two lines at equal distances from the central meridian will be projected without distortions. Any object not located on the surface of the reference ellipsoid will be also be subject to scale distortion caused by height, as shown in Figure 2.4.

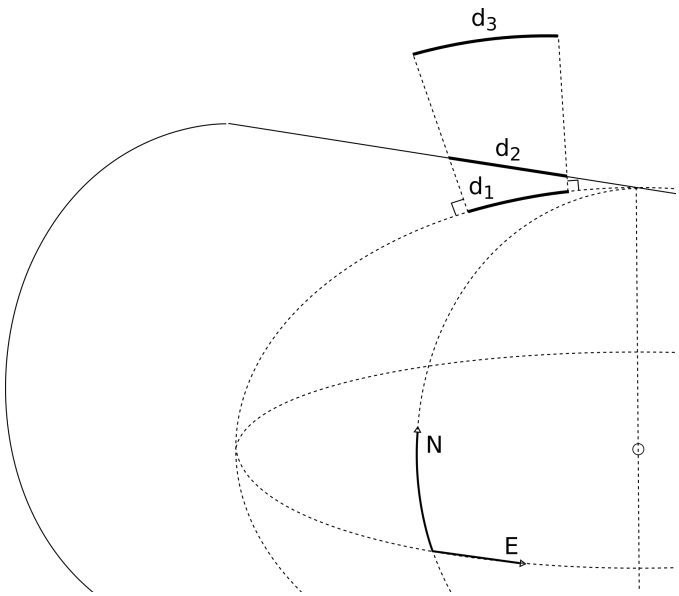

Figure 2.4: Difference in scale between three corresponding distances; d_1 on the reference ellipsoid, d_2 in the map projection, and d_3 in the terrain

In a tangent transverse Mercator projection, distortions are present for objects that are not tangent with the central meridian, and the amplitude of these distortions increase with the distance to the central meridian. The distortions caused by height over the reference ellipsoid and by distance to the central meridian act in opposite directions, and it is therefore possible to find an infinite number of lines that are parallel to the central meridian, and that are at non-zero heights, where there is zero scale distortion between the reference ellipsoid and the map plane. Scale distortions are expressed in terms of part per million (ppm), where one ppm equals a difference of 1 mm per 1 km.

Universal transverse Mercator (UTM) is a system that divides the Earth into a series of transverse Mercator projection zones (Buchroithner and Pfahlbusch, 2017). A scale factor of 0.9996 is applied to each zone, which effectively makes the map

cylinder secant to reference ellipsoid instead of tangent. The scale factor causes a scale distortion of 400 ppm along the central meridian of the map projection, but it also creates two lines that are parallel to the central meridian where there is no scale distortion.

Different CRSs and map projections are used for different areas, and they can be divided into the categories global and local. WGS 84 is a global CRS that is used for the Global Positioning System (GPS), and it is also commonly used together with UTM projections. In Sweden, SWEREF 99 is the official national CRS. The reference ellipsoids used by WGS 84 and SWEREF 99 are virtually identical, but positions in the two systems differed roughly 7-8 decimeters in 2017, and the difference is increasing a few centimeters per year due to continental drift (Lantmäteriet, 2017a). Sweden is divided into 12 zones that are 1.5° apart, and a local map projection based on SWEREF 99 are used for each zone. By using these local projection zones, scale distortion can be kept below 50 ppm for the vast majority of all locations in Sweden (Lantmäteriet, 2017b).

Object-specific map projections are map projections that are custom defined to produce very low scale distortions in a specific area, such as the extents of a construction project. Common projection types are transverse and oblique Mercator projections, where the oblique Mercator projection is a cylindrical map projection where the central line is not a meridian (Snyder, 1987). These types of map projections are very well suited for large longitudinal projects, such as roads and railroads, as the map projection can be defined in such a way that all objects of interest remain in a relatively narrow corridor around the central meridian or central line of the map projection. In this way, it is possible to keep scale distortions low, even over large areas.

An appropriate choice of map projection can minimize the scale distortions in a project area, but scale distortions of varying magnitude will always be present. In order to further reduce scale distortions, different regions within the project area have to be handled separately using a residual model, which is a model that applies local scale factors to compensate for local variations in scale distortions. An example of such an implementation is SnakeGrid (Iliffe et al., 2007), where an oblique Mercator projection is accompanied by a residual model in order to create a corridor along a curved road or railroad geometry where scale distortions are very low.

2.1.3 Local Cartesian coordinate systems

In the local Cartesian coordinate system, which is referred to as *engineering system* in Paper 1, it is assumed that there is a constant 1:1 scale between the engineering system and the terrain throughout the entire area. It is also assumed the any line parallel to the "up" axis in the engineering system is also parallel to the gravity vector in the terrain.

The engineering system can be transformed into another local Cartesian coordinate system that is oriented towards the geodetic North pole. This system is

from now on referred to as the *oriented engineering system*, see Figure 2.1, and it is identical to the engineering system except for its orientation. The transformation from the engineering system (x, y, u) to the oriented engineering system (e, n, u) is performed using (Hofmann-Wellenhof et al., 2008):

$$\begin{bmatrix} e \\ n \\ u \end{bmatrix} = \begin{bmatrix} cos\alpha & -sin\alpha & 0 \\ sin\alpha & cos\alpha & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ u \end{bmatrix} \tag{2.8}$$

where α is the rotation angle between the y-axis and the geodetic North.

2.1.4 Height systems

A height system is used to describe the height of a point relative to a reference surface, which typically is either a reference ellipsoid or a geoid[1]. A height system with a defined connection to the physical Earth is called a vertical datum. Height above a reference ellipsoid h and height above a geoid H are related to each other in the following way:

$$h = H + N \tag{2.9}$$

where N is the height difference between the geoid and the ellipsoid at the given location, which is given by the geoid model. It is not only the magnitude that differs between ellipsoidal and geoidal heights, but the "up" direction is different as well. Height relative to the reference ellipsoid is defined as perpendicular to the surface of the reference ellipsoid while height relative to the geoid is defined as perpendicular to the geoid, which is the same as the direction of gravity. The angle difference between the two height definitions is usually limited to 20 or 70 arc seconds for flat and mountainous terrain, respectively (Bomford, 1980), and given its small magnitude this difference is disregarded in this thesis.

2.2 IFC

This section gives a short description of IFC and its geographic components. For a more complete description, see Paper 1. IFC is based on STEP EXPRESS (ISO 10303), which is a data modeling language that provides machine readable and unambiguous descriptions of products (Kramer and Xu, 2009). IFC was in its beginning in 1994 more focused on the technical architecture of the format, but has since then gradually started to also specify the processes around the format as well (Laakso and Kiviniemi, 2012).

[1] A geoid is a surface based on the gravity field of the Earth that approximates the mean sea level

The geographic location and orientation of an engineering system can be stored in two different ways; as latitude, longitude, height above ellipsoid, and rotation relative the geodetic North, or as northing, easting, height in a vertical reference system, and rotation relative the cartographic North. The latitude, longitude, and height above ellipsoid are commonly used for energy computations where only an approximate location is required, and it is therefore not recommended to use these attributes for georeferencing. Northing, easting, height, and rotation relative to the cartographic North were a later addition to the standard with the specific aim to be used for georeferencing.

The IFC entity *IfcMapConversion* (buildingSMART, 2013), which defines the transformation from the engineering system to a map projection, contains the following attributes:

- SourceCRS
- TargetCRS
- Eastings
- Northings
- OrthogonalHeight
- XAxisAbscissa
- XAxisOrdinate
- Scale

SourceCRS and *TargetCRS* refer to the engineering system and the map projection, respectively, and the map projection is identified via its EPSG number, which is a global identifier for common map projections. *Eastings* and *Northings* specify the coordinates of the origin of the engineering system within the map projection, and *OrthogonalHeight* specifies the height of the origin of the engineering system in a vertical reference system. *XAxisAbscissa* and *XAxisOrdinate* together define the rotation angle between the x-axis in the engineering system and the E-axis in the map projection, which is the same as the angle between the y-axis in the engineering system and the cartographic North. *Scale* is a scale factor that affects all three coordinate axes.

2.3 Georeferencing local Cartesian coordinate systems

There are mainly two ways to georeference a local Cartesian coordinate system; by rotating it towards the geodetic North (see Equation 2.8) and relating it to the reference ellipsoid in 3D, or by splitting the local Cartesian coordinate system into

2D (horizontal) and 1D (vertical) and georeferencing the horizontal and vertical coordinates separately using a map projection and vertical datum. The first approach can be suitable for smaller areas such as single buildings, but for larger areas, and especially for large longitudinal projects, it is preferable to use a map projection.

Because there are scale distortions caused by height relative to the reference ellipsoid and by distance to the central meridian, it is necessary to apply a scale factor to compensate for these scale distortions. A correctly applied scale factor can minimize the scale distortions for a certain project, and a lower variation in scale distortion within the project area, the greater the effect of this reduction. It is therefore desirable to choose a map projection that produce close to constant scale distortions in the entire project area.

Paper 1 compares 3 different georeferencing methods that all are compatible with the IFC specification and where verticality is maintained throughout the transformation. This means that any line that is perpendicular to the horizontal plane in the engineering system will be perpendicular to the reference ellipsoid after transformation.

The methods evaluated in Paper 1 are:

Method 1 The engineering system is split into its horizontal and vertical components, where the horizontal components are georeferenced using a map projection and the vertical components are redefined as height above ellipsoid. No horizontal scaling is applied as there is no suitable scale factor in IFC.

Method 2 The engineering system is related to the reference ellipsoid in 3D. In order to maintain the verticality of lines, all points within the engineering system are georeferenced as if they had a height coordinate of 0.

Method 3 A local reference ellipsoid is created based on the latitude and height of the project area. The horizontal components of the engineering system are georeferenced by transforming them to polar coordinates and by computing the corresponding geodetic lines on the local ellipsoid, and the vertical components are redefined as height above the local ellipsoid. The geometries are transformed from the local ellipsoid to the reference ellipsoid via ECEF coordinates.

For in-depth descriptions of the methods including equations, see Paper 1. The results show that the scale distortion caused by Method 1 greatly depends on the geographical location of the project and the map projections that are available for that area, while Method 3 performs equally well in all locations. Method 2 performed worse than Methods 1 and 3 for all locations. If support for horizontal scaling and object specific map projections was included in IFC, Method 1 would perform better than Method 3 for all locations.

2.4 Limitations of IFC and suggested improvements

This section is a summary of the findings in Paper 1. The following limitations in the IFC standard were identified:

- Lack of support for horizontal scaling
- Lack of support for object-specific map projections
- Lack of support for residual models

In Paper 1 it is shown that proper use of a horizontal scale factor can at a minimum reduce the scale distortions between the engineering system and reference ellipsoid by half, and for large longitudinal projects that are parallel to central meridian of a map projection, and at a constant height, the horizontal scaling can eliminate virtually all scale distortions. A horizontal scale factor can be seen as a tool that can fit the geometries in the engineering to a given map projection in an optimum manner.

There are scenarios, e.g. when constructing large longitudinal projects across a projection zone, when the use of a horizontal scale factor will not be enough to keep scale distortions at low enough levels. A solution can in these cases be to use an object-specific map projection, such as an oblique Mercator projection. If the use of horizontal scaling allows the geometries to be fitted to the map projection in an optimum manner, the use of object specific map projections allows that allows that the map projection is optimally fitted to the Earth, considering the specific project area.

Two sources of scale distortion not handled by horizontal scaling and object-specific map projections are varying distance to the central line or central meridian, and varying height above the ellipsoid. In order to eliminate these types of distortions, it is necessary to handle different regions within the project area differently, compensating for local variations in terms of scale distortion. This can be done using a residual model, which applies different scaling to different areas.

Implementing support for a horizontal scale factor and object-specific map projections in IFC would only require the addition of two attributes to the entity *IfcMapConversion*; one string and one scalar. Residual models are of a more complex datatype that cannot easily be expressed as a numeric value or as a string, and would therefore require further consideration. However, the addition of a horizontal scale factor and support for object-specific map projections would make sure that IFC meets the needs of a wide range of road and railroad construction projects.

2.5 Inverse georeferencing

Paper 1 also recognizes the fact that any transformation applied to geometries being exported from the BIM environment must by inversely applied to the geometries being imported to the BIM environment. For example, consider a road from A

to B where the design is based on a terrain model that is georeferenced in a map projection. The designer assumes that distances in the engineering system will be distances on the ground and designs the road accordingly. The road is transformed and scaled to a map projection and then finally constructed in the terrain. Whether the terrain model is inversely transformed when imported into the design environment or not has an impact on the outcome, as shown in Figure 2.5.

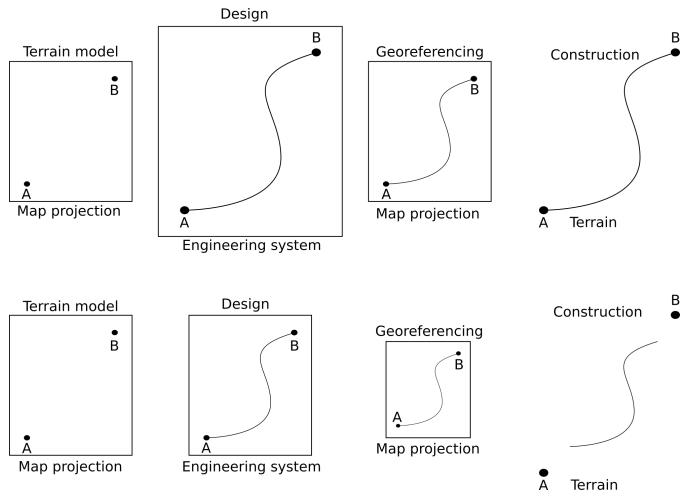

Figure 2.5: Series of images illustrating the consequences of not applying inverse georeferencing. The label above each image indicates the phase in the construction process, and the label below each image indicates the coordinate system used in this phase. The upper row shows a sequence where the underlying terrain model is inversely georeferenced when imported to the design environment, and the lower row shows a sequence where inverse georeferencing is not used.

These types of distortions might be small for one individual transformation but they will accumulate over time. In autonomous information systems where data may be frequently imported and exported, it is paramount that this can be done without compromising the quality of the data.

One could use an approach where the terrain model is not inversely transformed when imported to the design environment, and the designed road is not transformed before its staked out in the terrain. By doing so, the constructed road would still span the entire distance from A to B. However, this approach will cause distortions in the dimensions of the road as illustrated in Figure 2.6.

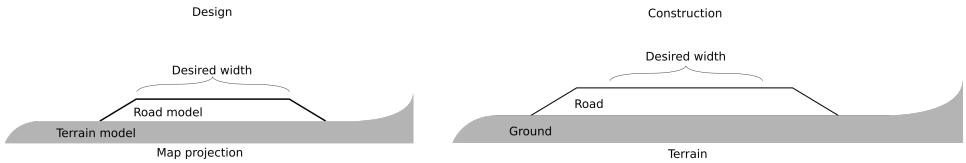

Figure 2.6: Cross sections of a road in the design and construction phases. The road is designed directly in the map projection and is not transformed before it is staked out in the terrain. Using this approach, the road will span the correct distance, but the dimensions of the constructed road will not be the same as the dimensions of the designed road. The labels above the images indicate the phase in the construction process, and the labels below the images indicate the corresponding coordinate systems.

As illustrated in Figures 2.5 and 2.6, designing directly in a map projection or failing to apply inverse transformations leads to discrepancies between the designed and the constructed geometries. Paper 1 shows that no georeferencing method is free from distortions, but by choosing an appropriate map projection for the project area, conducting design in a local Cartesian coordinate system, and by applying correct transformations in both directions, these distortions can be kept at a minimum. It also ensures that the quality of the data is not degraded by repeated imports and exports.

Chapter 3

Object identification in point clouds

Point clouds, created either by laser scanning or by photogrammetry, contain large volumes of georeferenced 3D data. Point clouds give a more complete description of a scene compared to individual total station or global navigation satellite system (GNSS) measurements, and are therefore commonly used for object identification and as a basis for creating BIM models (Thomson et al., 2013; Xiong et al., 2013; Thomson and Boehm, 2015).

A laser scanner emits laser pulses and records the returns, and since the direction of the pulse and the elapsed time between the emission and the return are known, it is possible to calculate the location of the object that caused the return. This is repeated in a scanning fashion and the observations are combined into a 3D scene. Photogrammetry utilizes two or more images covering the same object to compute the location of points that are visible in both images. This requires that the intrinsic[1] and extrinsic[2] camera parameters are either known or can be estimated. The resulting data set is from both methods an unstructured list of 3D points, typically accompanied by RGB colors. A point cloud created by a laser scanner would also include the intensity of the returning pulse and the timestamp for each point. The RGB colors in a point cloud produced by laser scanning are captured by cameras mounted close to the scanner, and the color is transferred to the point cloud by using perspective projection.

Automatic feature extraction and object reconstruction has been recognized as an important enabler for a more wide-spread use of BIM for existing assets (Volk et al., 2014), and it is equally useful for creating as-built models for newly constructed assets. Papers 2 and 3 explore the possibilities to use image-based deep learning to identify and extract objects from points clouds. The principles of deep

[1] The intrinsic camera parameters are the focal length, the radial distortion, and the intersection of the image sensor and the optical axis

[2] The extrinsic camera parameters are the location and the orientation of the camera

learning are explained in Section 3.1.

3.1 Deep learning and convolutional neural networks

The core operation in deep learning is the transformation of one vector, the input, to another vector, the output, by passing it through a deep neural network, where each layer performs a non-linear transformation of the vector (Goodfellow et al., 2016). The most straight-forward network architecture used is called a multilayer perceptron (MLP), which consists of a minimum of an input layer, a hidden layer, and an output layer, and where all nodes in a layer are connected to all nodes in the next layer. The non-linear transformation allows such a network to solve problems that are not linearly separable, such as the XOR problem[3]. An algebraic representation of an MLP with two hidden layers would be:

$$f(x) = f_3(f_2(f_1(x))) \tag{3.1}$$

where $f(x)$ is the MLP, and f_{1-3} are the functions of the two hidden layers and the output layer.

An MLP is trained using supervised learning, where an input is passed through the network and a loss is computed by comparing the output of the network to the label of the input. The network learns through backpropagation, where gradients for all parameters are computed from the loss using the chain rule (LeCun et al., 2015). The parameters in the network are then updated according to their gradients. Training a neural network in this way requires training data, which consists of sample inputs accompanied by the correct label.

The convolutional neural network (CNN), pioneered by LeCun et al. (1990), is a neural network designed specifically for the processing of grid-like data such as images, which can be seen as 2D grids (Goodfellow et al., 2016). The contribution of Krizhevsky et al. (2012) to the 2012 ImageNet Large Scale Visual Recognition Challenge (ILSVRC) (Russakovsky et al., 2015), cemented the CNN as the top-performing architecture for all vision related recognition tasks (LeCun et al., 2015). The main operator in a CNN is the convolutional layer, in which a filter processes all locations in the image independently from each other.

The concept of transfer learning (Donahue et al., 2013; Razavian et al., 2014) is based on the recognition of the fact that most parameters in a CNN are used to extract geometric primitives of varying complexity, and that any real semantic identification is taking place in the topmost layers. This means that a network trained on e.g. the ImageNet data set, which consists of over 1.4 million images belonging to 1000 different classes (Russakovsky et al., 2015), can be repurposed to

[3]The XOR problem is a common example used in classification. It can be visualized by considering four points in 2D, located at $(0,0), (0,1), (1,0)$, and $(1,1)$, where the first and the last belong to class a and the other two belong to class b. No single line can separate a from b, and no linear transformation of the 2D space can change this. However, a non-linear transformation of the 2D space can make the classes linearly separable (Goodfellow et al., 2016).

another semantic domain by replacing and retraining the topmost layers on a new data set. The layers that are kept from the previous training can either be frozen, effectively turning them into a static feature extractor, or just initialized with their learned weights while still being subject to further training. The data set used to repurpose the network can be several orders of magnitude smaller than ImageNet and the network will still maintain most of its performance.

The fully convolutional network (FCN) is a development of the CNN that instead of classifying entire images classifies individual pixels, which is a procedure that is also known as semantic segmentation. The FCN was developed by Long et al. (2015), and only differs from a conventional CNN in the topmost layer. The top layer in a conventional CNN is a fully-connected layer that produces an output vector with as many elements as there are classes. In an FCN, this layer is replaced by yet another convolutional layer of size 1×1, and with a depth corresponding to the number of classes. Due to the fact that all layers below the top remain unchanged, it is possible to use a network trained for image recognition on e.g. the ImageNet data set, change the top layer turning it into an FCN, and still utilize what the network learned from its previous training.

3.2 Applications for semantic segmentation in geodata processing

The use of deep learning is today relatively common in the field of remote sensing, where it for example is used to classify satellite imagery (Zhu et al., 2017). The use of deep learning to classify point clouds is less researched, and the approaches are not as straight-forward. For related work, see Papers 2 and 3.

Papers 2 and 3 show how semantic segmentation can be used to identify and extract objects from points clouds created by a road-borne MMS equipped with laser scanners and cameras. The pixels in the images are classified by using an FCN to perform semantic segmentation, and the pixel classification is then transferred to the point cloud by projecting the point cloud to the image plane:

$$\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} \cdot \frac{1}{Z} \tag{3.2}$$

where x and y are coordinates in the image plane, and where X, Y, and Z are coordinates in the point cloud. Pixel coordinates are computed for all points using:

$$\begin{bmatrix} u \\ v \end{bmatrix} = \begin{bmatrix} f_u & 1 & c_u \\ 0 & f_v & c_v \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \tag{3.3}$$

where u and v are pixel coordinates, f_u and f_v are the camera's focal lengths, and c_u and c_v are the pixel coordinates where the optical axis intersect the image sensor.

This methodology can not only be used for point clouds created by laser scanning, but also for point clouds created by photogrammetry. In the case of photogrammetry, there is no need for perspective projection, as there is a topological connection between each point and its corresponding pixels. This also eliminates the problem of occlusion that is present when working with perspective projection. Because it is unknown whether a point is visible in an image, all points in the camera field of view are projected to the image plane, even though some of them might be occluded by other objects. In the case of photogrammetry there is no such problem, as all points were created from pixels and therefore have to be visible in their corresponding images.

The now classified point cloud requires some sort of geometric post-processing in order to extract the points corresponding to individual objects. The type of post-processing required will vary depending on the object type. In the examples in Papers 2 and 3, the object type noise barrier only requires a minimum of post-processing while the object type game fence require an elaborate algorithm.

3.2.1 Combining different types of data in the neural network

The methodology proposed in 3.2 only utilizes the RGB images in the semantic segmentation, and since the point cloud contains geometry and possibly also intensity, this leaves information unused that could potentially improve the results of the semantic segmentation. It is possible to create depth and intensity maps by projecting the point cloud to the image plane. These maps could in turn be added to to the red, green, and blue bands of the image and create a 5-band image that includes the geometry and intensity of the point cloud. It is possible to design an FCN that accepts 5-band images as its input, but doing so would make it impossible to utilize transfer learning as no large data sets of 5-band images are available. Also, since the results from the semantic segmentation presented in 3.3 are relatively high, the potential benefit of including geometry and intensity is relatively limited. It could prove useful for object types that have very similar RGB texture but very different geometry or intensity, but such objects could very likely be distinguished from each other regardless in the post-processing stage.

3.3 Implementations for identifying noise barriers, road barriers, and game fences in mobile mapping data

Paper 2 describes an implementation of the methodology presented in Section 3.2 where it is used to identify noise barriers in mobile mapping data, and Paper 3 describes two implementations where the same methodology is used to identify road barriers and game fences.

The data set used for all three implementations covers 7 km of country road outside Mariestad in Västergötland, Sweden. It includes a point cloud, consisting

of roughly 160 million points, and roughly 4200 images captured at intervals of 0.72 s and in 6 directions.

The images were cropped into tiles of size 1000×1000 pixels, and from these, pools of tiles both containing and not containing the objects of interest were created. An FCN was trained repeatedly using varying amounts (up to 80%) of the available training data, and the trained network was validated against the remaining portion. The performance metrics precision, recall, and Cohen's Kappa coefficient (κ) (Cohen, 1960) were computed for each training session:

$$Precision = \frac{TP}{TP + FP} \tag{3.4}$$

$$Recall = \frac{TP}{TP + FN} \tag{3.5}$$

$$\kappa = \frac{p_o - p_e}{1 - p_e} \tag{3.6}$$

where TP is true positive, FP is false positive, FN is false negative, p_o is observed agreement, and p_e is chance agreement. Precision indicates how many of the identified pixels were relevant, recall indicates how many of the relevant pixels that were identified, and Cohen's Kappa coefficient is a comparison between the actual classification and a random classification.

The trained network was then used to create predictions, images where the pixel value indicate class affiliation, and this classification was then transferred to the point cloud as described in Section 3.2. Manual classifications of the point cloud were performed for all three object types, and precision and recall were computed between the automatically extracted points and the manual classifications.

3.3.1 Noise barriers

The training data in the noise barrier implementation consisted of 96 tiles containing noise barrier pixels and 360 tiles not containing noise barrier pixels. The training was performed with and without data augmentation in the form of horizontal mirroring, and the results showed that the mirroring effectively doubled the amount of training data without increasing the manual effort.

The average precision, recall and Kappa coefficient for a network trained on 80% of the available training data, using data augmentation, were 96.9%, 91.5%, and 0.945, respectively. The resulting point cloud was post-processed geometrically by dividing it into clusters and removing noise based on spatial proximity, where one cluster represented one noise barrier object.

A manual classification of the point cloud was performed and the result was used as ground truth. The precision and recall for the automatically extracted points compared to the ground truth were 93.5% and 95.0%, respectively, before the geometric post-processing, and the corresponding numbers were 97.1% and 91.5% after the geometric post-processing.

3.3.2 Road barriers

The training data in the road barrier implementation consisted of 232 tiles containing road barrier pixels and 202 tiles not containing road barrier pixels. The training samples were augmented by horizontal mirroring, and the average precision, recall, and Kappa coefficient for a network trained on 80% of the available training data were 93.9%, 91.0%, and 0.925, respectively.

Only the pixels constituting the actual barriers were labeled in the training data, and the supports, poles, and other features were left out due their thin geometries. In order to include these features, it was necessary to expand the selected subset of points. Ground points were filtered from exported point cloud, and the remaining points were used as input in a query selecting all points within a certain distance from the original point cloud. This query made sure the missing features were included, and ground points and noise were filtered from the resulting point cloud.

The results were compared to a manually selected ground truth, and the precision and recall before post-processing were 75.7% and 69.5%, respectively, and 88.2% and 86.6% after post-processing. The fact that the subset was expanded made it possible to increase both the precision and the recall.

3.3.3 Game fences

The training data in the game fence implementation consisted of 672 tiles containing game fence pixels and 531 tiles not containing game fence pixels. The training samples were augmented by horizontal mirroring, and the average precision, recall, and Kappa coefficient for a network trained on 80% of the available training data were 93.3%, 85.5%, and 0.886, respectively.

The post-processing of game fences required a much more elaborate procedure than the noise barriers and road barriers. Due to their transparent nature, all objects behind the game fence, such as trees, were captured in the point cloud, and because the classification was transferred using perspective projection, all points behind the game fence were classified as game fence as well. It was therefore necessary to find a way to separate the game fence points from e.g. tree points.

One characteristic that separates game fences from trees is that they, when viewed from above, form a poly-line consisting of linear segments. These segments were identified using Hough transform (Ballard, 1981), which is an algorithm that is able to identify weak geometric shapes in images with high noise ratios. The Hough transform was used to identify the line consisting of the most points, and any points along the extents of the line were removed from the point cloud. This procedure was repeated until there were no large enough lines left to detect. A poly-line was formed out of the linear segments by fitting linear equations to each point cluster and by intersecting neighboring lines. The order of the segments was known due to the timestamps of points in the point cloud. The poly-line was then matched with the initial point cloud, where all ground points had been removed,

and the final game fence classification consisted of the points in the point cloud that coincided with the poly-line in the horizontal plane.

The precision and recall compared to a manual classification were before post-processing 11.4% and 100.0%, respectively, and 78.3% and 83.6% after post-processing. The point cloud included areas where the point density was significantly lower, and where only the poles from the game fences were visible. Due to this fact, a subset of the point cloud where all game fences were clearly visible was selected, and the classification was repeated for this subset. The precision and recall for the subset were 13.6% and 100.0% before post-processing, and 88.0% and 99.5% after post-processing.

Chapter 4

Conclusions and outlook

Higher degrees of automation require that machines autonomously can interpret and process data without the aid of humans. In the case of integrating BIM models with geodata, it is of great importance that all information required to perform the necessary transformations is available in a machine readable and unambiguous format. IFC has taken a step towards this with the addition of the entity *IfcMap-Conversion*, which describes the transformation from the engineering system to a map projection, but the transformation parameters are lacking.

Paper 1 shows that proper horizontal scaling can reduce scale distortion by at least half for any project. The benefit of using an object-specific map projection compared to a conventional map projection depends on the geographic location of the project and the conventional map projections available in that area. A road constructed across a projection zone would be subject to scale distortions up to 50 ppm using a local SWEREF 99 projection, or up to 400 ppm using a UTM projection. Such scale distortions could, depending on the geometry and extent of the particular road, be close to eliminated by using an object-specific map projection. The addition of support for horizontal scaling and object-specific map projection to IFC would make the standard suitable for a wider range of large longitudinal construction projects.

In the same way as the georeferencing of BIM models should be a strict and unambiguous process, the use of geodata in the design environment should be strict and unambiguous as well. The georeferencing of point clouds is today not as standardized as the georeferencing of total station measurements, and the choice of georeferencing method has implications for how the point cloud can be used, as shown in Section 2.5. These implications propagate throughout all of the design process. Future research should describe the implications of the different georeferencing procedures, and suggest proper methods for importing different types of point clouds into the engineering system.

Papers 2 and 3 show how semantic segmentation of images in combination with transfer learning can be used successfully to identify semantic objects in point

clouds. The proposed methodology achieves over 85% in both precision and recall compared to manual approaches for all three of the tested object types. The possible applications of the proposed methodology extend beyond the implementations shown in Papers 2 and 3, and include point clouds produced by both laser scanning and photogrammetry, as well as all types of platforms and terrestrial setups. Further investigation of this methodology could include studies with different data sets, object types, and network architectures. Future research should also investigate the applicability of using the PointNet and PointNet++ architectures (Qi et al., 2017a,b), which are specifically designed for end-to-end point cloud segmentation, and compare their performance to the methodology proposed in Papers 2 and 3. Another topic that is of interest is the development of efficient network architectures that can utilize all the data types captured by photogrammetry and laser scanning, but as discussed in Section 3.2.1, this area is limited due to low availability of already classified training data.

This thesis has identified two clear intersections between the fields of BIM and geodesy, and it has made contributions to the technical development in both these. Geographic data and metadata are vital parts for both the construction and maintenance of the built environment, and they need to be integral parts of any information system concerning these domains. The field of geodesy and surveying is responsible for providing other actors in the industry with geodata, and the adoption of BIM and the digitalization of the urban environment are changing the demands. It is growing increasingly important to be able to derive semantic information from the captured geodata, and the developments in the field of deep learning are giving geodesists new possibilities to do so.

Additional contributions

Presentation Harrie, L., Sun, J. and Uggla, G., *Forskningsprojekt*, 2018, Kartdagarna, Linköping, Sweden

Paper Uggla, G. and Horemuz, M., *Georeferencing methods for IFC*, 2018, Baltic Geodetic Congress, Olsztyn, Poland

Presentation Uggla, G., *Bildbaserad objektidentifiering i punktmoln*, 2019, Geodesidagarna, Göteborg

References

Azhar, S. (2011). Building information modeling (bim): Trends, benefits, risks, and challenges for the AEC industry. *Leadership and Management in Engineering*, 11(3):241–252.

Ballard, D. (1981). Generalizing the hough transform to detect arbitrary shapes. *Pattern Recognition*, 13(2):111–122.

Bomford, G. (1980). *Geodesy, 4th edition*. Oxford University Press, Oxford.

Boschert, S. and Rosen, R. (2016). Digital twin—the simulation aspect. In Hehenberger, P. and Bradley, D., editors, *Mechatronic Futures: Challenges and Solutions for Mechatronic Systems and their Designers*, pages 59–74. Springer International Publishing, Cham.

Buchroithner, M. F. and Pfahlbusch, R. (2017). Geodetic grids in authoritative maps – new findings about the origin of the UTM grid. *Cartography and Geographic Information Science*, 44(3):186–200.

buildingSMART (2013). Industry Foundation Classes release 4 - addendum 2 - reference view (IFC4 RV 1.1). http://www.buildingsmart-tech.org/ifc/review/IFC4Add2/ifc4-add2-rv/html/schema/templates/project-global-positioning.htm. Last accessed 2019-02-20.

Cohen, J. (1960). A coefficient of agreement for nominal scales. *Educational and Physiological Measurement*, 20(1):37–46.

Delbrügger, T., Lenz, L. T., Losch, D., and Roßmann, J. (2017). A navigation framework for digital twins of factories based on building information modeling. In *2017 22nd IEEE International Conference on Emerging Technologies and Factory Automation (ETFA)*, pages 1—4.

Donahue, J., Jia, Y., Vinyals, O., Hoffman, J., Zhang, N., Tzeng, E., and Darrell, T. (2013). Decaf: A deep convolutional activation feature for generic visual recognition. arXiv:1310.1531 [cs.CV].

Eastman, C., Teicholz, P., Sacks, R., and Liston, K. (2011). *BIM Handbook: A Guide to Building Information Modeling for Owners, Managers, Designers, Engineers and Contractors, 2nd Edition.* John Wiley & Sons, Inc., Hoboken, New Jersey.

Goodfellow, I., Bengio, Y., and Courville, A. (2016). *Deep Learning.* MIT Press. http://www.deeplearningbook.org, last accessed 2019-02-01.

He, K., Zhang, X., Ren, S., and Sun, J. (2016). Deep residual learning for image recognition. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778.

Hofmann-Wellenhof, B., Lichtenegger, H., and Wasle, E. (2008). *GNSS – Global Navigation Satellite Systems.* Springer-Verlag Wien, 1 edition.

Iliffe, J. C., Arthur, J. V., and Preston, C. (2007). The Snake Projection: A Customised Grid for Rail Projects. *Survey Review*, 39(304):90–99.

Kramer, T. and Xu, X. (2009). STEP in a Nutshell. In Xu, X. and Nee, A., editors, *Advanced Design and Manufacturing Based on STEP. Springer Series in Advanced Manufacturing.* Springer, London.

Krizhevsky, A., Sutskever, I., and Hinton, G. E. (2012). ImageNet classification with deep convolutional neural networks. In *Proceedings of the 25th International Conference on Neural Information Processing Systems - Volume 1*, NIPS'12, pages 1097–1105, USA. Curran Associates Inc.

Laakso, M. and Kiviniemi, A. (2012). The IFC standard - A review of history, development, and standardization. *Electronic Journal of Information Technology in Construction*, 17(May):134–161.

Lantmäteriet (2017a). SWEREF 99. https://www.lantmateriet.se/sv/Kartor-och-geografisk-information/GPS-och-geodetisk-matning/Referenssystem/Tredimensionella-system/SWEREF-99. Last accessed 2019-03-12.

Lantmäteriet (2017b). SWEREF 99, map projections. https://www.lantmateriet.se/sv/Kartor-och-geografisk-information/GPS-och-geodetisk-matning/Referenssystem/Tvadimensionella-system/SWEREF-99-projektioner. Last accessed 2019-03-12.

LeCun, Y., Bengio, Y., and Hinton, G. (2015). Deep learning. *Nature*, 521:436–444.

LeCun, Y., Boser, B., Denker, J. S., Howard, R. E., Habbard, W., Jackel, L. D., and Henderson, D. (1990). Advances in neural information processing systems 2. chapter Handwritten Digit Recognition with a Back-propagation Network, pages 396–404. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA.

Long, J., Shelhamer, E., and Darrell, T. (2015). Fully convolutional networks for semantic segmentation. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3431–3440.

Qi, C. R., Su, H., Kaichun, M., and Guibas, L. J. (2017a). Pointnet: Deep learning on point sets for 3d classification and segmentation. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 77–85.

Qi, C. R., Yi, L., Su, H., and Guibas, L. J. (2017b). PointNet++: Deep hierarchical feature learning on point sets in a metric space. In Guyon, I., Luxburg, U. V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., and Garnett, R., editors, *Advances in Neural Information Processing Systems 30*, pages 5099–5108. Curran Associates, Inc.

Razavian, A., Azizpour, H., Sullivan, J., and Carlsson, S. (2014). CNN features off-the-shelf: An astounding baseline for recognition. In *2014 IEEE conference on computer vision and pattern recognition workshops*, volume 1403.

Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., Berg, A. C., and Fei-Fei, L. (2015). ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)*, 115(3):211–252.

Snyder, J. P. (1987). *Map Projections: A working manual*. U.S. Government Printing Office, Washington, D.C.

Succar, B. (2009). Building information modelling framework: A research and delivery foundation for industry stakeholders. *Automation in Construction*, 18:357–375.

Thomson, C., Apostolopoulos, G., Backes, D., and Boehm, J. (2013). Mobile laser scanning for indoor modelling. *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences*, II-5/W2:289–293.

Thomson, C. and Boehm, J. (2015). Automatic geometry generation from point clouds for BIM. *Remote Sensing*, 7(9):11753–11775.

Volk, R., Stengel, J., and Schultmann, F. (2014). Building information modeling (BIM) for existing buildings — literature review and future needs. *Automation in Construction*, 38:109–127.

Xiong, X., Adan, A., Akinci, B., and Huber, D. (2013). Automatic creation of semantically rich 3D building models from laser scanner data. *Automation in Construction*, 31:325–337.

Yan, H. and Demian, P. (2008). Benefits and barriers of building information modelling. In Ren, A., Ma, Z., and Lu, X., editors, *12th International Conference on Computing in Civil and Building Engineering*, volume 161, Beijing, China.

Zhu, X. X., Tuia, D., Mou, L., Xia, G., Zhang, L., Xu, F., and Fraundorfer, F. (2017). Deep learning in remote sensing: A comprehensive review and list of resources. *IEEE Geoscience and Remote Sensing Magazine*, 5(4):8 – 36.