Generating, storing, updating and disseminating a countrywide 3D model

Dukai, B.; Peters, R.; Wu, T.; Commandeur, T.; Ledoux, H.; Baving, T.; Post, M.; Van Altena, V.; Van Hinsbergh, W.; Stoter, J.

Citation (APA)
Dukai, B., Peters, R., Wu, T., Commandeur, T., Ledoux, H., Baving, T., Post, M., Van Altena, V., Van Hinsbergh, W., & Stoter, J. (2020). Generating, storing, updating and disseminating a countrywide 3D model. *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences - ISPRS Archives*, *44*(4/W1), 27-32. https://doi.org/10.5194/isprs-archives-XLIV-4-W1-2020-27-2020

Important note
To cite this publication, please use the final published version (if applicable).
Please check the document version above.

Generating, storing, updating and disseminating a countrywide 3D model

Balázs Dukai[1], Ravi Peters[1], Teng Wu[1], Tom Commandeur[1], Hugo Ledoux[1], Tony Baving[2], Marc Post[2], Vincent van Altena[2], Willem van Hinsbergh[2], Jantien Stoter[1,2*]

[1]3D geoinformation group, Department of Urbanism, Faculty of the Built Environment, Delft University of Technology, Julianalaan 134 Delft 2628BL the Netherlands
{b.dukai|r.y.peters|t.wu|t.j.f.commandeur|h.ledoux|j.e.stoter}@tudelft.nl

[2]Kadaster, Hofstraat 110, 7311 KZ Apeldoorn, the Netherlands
{marc.post|tony.baving| vincent.altenavan|willem.vanHinsbergh@kadaster.nl|jantien.stoter}@kadaster.nl

KEY WORDS: 3D modelling, NMAs, 3D reconstruction

ABSTRACT: As in many countries, in The Netherlands governmental organisations are acquiring 3D city models to support their public tasks. However, this is still being done within individual organisation, resulting in differences in 3D city models within one country and sometimes covering the same area: i.e. differences in data structure, height references used, update cycle, data quality, use of the 3D data etc. In addition, often only large governmental organisations can afford investing in 3D city models (and the required knowledge) and not small organisations, like small municipalities. To address this problem, the Dutch Kadaster is collaborating with the 3D Geoinformation research group at TU Delft to generate and disseminate a 3D city model covering the whole of the Netherlands and to do this in a sustainable manner, i.e. with an implementation that ensures periodical updates and that aligns with the 3D city models of other governmental organisations, such as large cities. This article describes the workflow that has been developed and implemented.

1. INTRODUCTION

The use of 3D city models to address environmental challenges in urban areas has become common practice. Through recent advances in technologies to acquire 3D elevation information practitioners are able to automatically reconstruct 3D city models and use them in the fields of city planning and environmental simulations. However, current 3D city models produced by different organisations can differ a lot because of differences in acquisition methods, differences in applications for which the 3D data is collected, differences in data structures and formats etc. In addition, typically there is no plan to maintain and update the once generated data. Consequently, 3D city models are not part of governmental data infrastructure solutions and still not widely used in governmental decision processes.

To address this problem, the Dutch Kadaster (which has the public task to provide geo-information for common use) is collaborating with the 3D Geoinformation research group at TU Delft to (1) generate and disseminate a 3D city model containing 3D topography covering the whole of the Netherlands; and (2) to do this in a sustainable manner, i.e. with an implementation that ensures periodical updates and that aligns with the 3D city models of other governmental organisations, such as cities.

In this collaboration, a workflow is being developed that covers the different aspects varying from automated reconstruction from existing countrywide data, maintaining the 3D data in a seamless database, quality control and making the data available in an open 3D standard to be disseminated via the national governmental geoportal (PDOK.nl).

In this paper we describe the details of the workflow that combines several of our past researches and pilots.

It should be noted that National Mapping Agencies all over the world are generating, maintaining and disseminating 3D topography data. Initiatives in Europe are for example described in Stoter et al (2014; 2016).

2. SCOPE

The Netherlands has a framework of key registers, in which specific governments are responsible to collect specific data and other governments are obliged to use the data that is collected by other governmental organisations. At this moment, there are ten Key Registers (Digital Government, 2020). Topographical data about roads, water, land use, bridges, buildings etc is defined both in the Base register large-scale topography (*Basisregistratie Grootschalige Topografie, BGT*) and Base Register Topography (BRT, for scales 1:10k and smaller). In addition, information (including geometry) about buildings is part of the Base Register Addresses and Buildings (BAG).

All the three registers describing topography only contain (and prescribe) 2D data, although the data model for BGT supports the optional extension to 3D (Stoter et al, 2013; Brink et al 2012; 2013).

For the 3D topography that is topic of this paper, aligning with key registers containing topography is a prerequisite to be embedded in mainstream governmental information infrastructures.

Kadaster is responsible for the production and maintenance of the BRT. The largest scale of BRT is data at scale 1:10k and therefore less appropriate for 3D data reconstruction. However, the BRT covering the whole of the Netherlands is available since 2005. Before the availability of BAG and BGT it was the only countrywide dataset available for the Netherlands. Therefore, a first 3D model covering the whole country has been reconstructed in 2013 based on BRT-data (Oude Elberink et al, 2013). The work in this paper improves those results with additional challenges brought by the very high detail of BGT/BAG compared to BRT. In addition, the 2013-version was only generated: it was never maintained,

updated or provided in a standardised way as in the current project.

The acquisition of BGT and BAG data is the responsibility of many organisations that have a task to maintain public space (municipalities, provinces, waterboards, etc). Ideally, 3D data about large scale topography should be collected as part of the BGT and BAG and be the responsibility of all these different source holders. But at the moment it is not feasible for all these organisations - specifically the small ones - to acquire, model and maintain 3D data. Therefore, for the time being, the Kadaster will automatically reconstruct the 3D data as a derived product of the existing BGT/BAG data sets.

For the height data, the national height model of the Netherlands is used (AHN, 2020). This is a point cloud acquired by airborne lidar systems. In addition, points from dense image matching are used as up-to-date height data required for periodical updates of the 3D topographical data.

2.1 Content of 3D Model NL

The 3D base data set, the data set that we are describing in this paper, consists of three data products that are all automatically generated.

The first product, called *3D Basisbestand Volledig*, consists of BGT-terrain surfaces with buildings integrated in the terrain. The (3D) surfaces represent land use objects that together form the bare earth (roads, water, vegetation coverages) with additional surfaces for multilevel crossings (bridges). The volumetric building models in this first product are generated from the BAG building footprints by a simple extrusion to a single height, i.e. the so called LoD1.2 representation (see Biljecki et al. 2016).

The second product, *3D Basisbestand Gebouwen*, contains LoD1.3 representations of buildings, which are buildings represented as block models with different heights in case buildings have significant height jumps like a church with a tower or a house with a shed attached (Stoter et al, 2020). The footprint of the building models (both LoD1.2 and LoD1.3) is always set to the lowest neighbouring terrain point to prevent buildings floating above the terrain.

The third product - building height statistics *3D Hoogtestatistieken Gebouwen* - contains the 2D BAG geometries of buildings with several height values assigned to it. These different height values represent different reference heights based on different statistical parameters calculated for elevation points that fall within the building footprint. Dependent on the application, a user can decide which reference height to use to extrude the building footprint from this data set. This product contains information for both LoD1.2 and LoD1.3 representations.

2.2 Updates

To keep the 3D model up-to-date, the 3D model will periodically (once per year) be generated based on new versions of BAG and BGT data. The latter datasets have an actuality of 6 months. For up-to-date height data needed for mutated objects, we use a countrywide point cloud obtained from aerial images that are acquired every year. These heights are used for building objects that are newer than the LiDAR point cloud at the reference date of the reconstruction.

To identify buildings that are newer than the LiDAR (AHN3) points at the reference date, we compare the timelines of each BAG building before and after the date that AHN3 was acquired.

If these two timelines of one building are the same, we assume the building to be unchanged since AHN3 was acquired and these are reconstructed based on AHN3. For all other buildings, we compare the geometry at the two moments (i.e. before and after acquisition of AHN3) to see if the change was significant. The comparison is done by identifying the area of difference. If this is smaller than a certain threshold (e.g. 2m), also these buildings are assigned to the collection of unchanged buildings. Consequently, all unchanged buildings are reconstructed from LiDAR point clouds and the changed buildings from point clouds obtained from dense image matching.

3. METHODOLOGY

To generate the data products as described in the previous section we have developed a methodology from reconstruction to dissemination. This workflow consists of the following steps that will be further detailed in the remaining subsections:

1. Pre-processing BAG/BAG
2. 3D reconstruction of the surfaces covering the bare earth
3. Reconstruction of 3D representation of buildings including quality parameters
4. LoD1.3 reconstruction by a method that first reconstructs LoD2 representations and then generalises them to LoD1.3
5. Making the data available in the CityJSON standard for dissemination
6. Identifying and solving multi-level building situations, since the BAG does not contain information on such situations.
7. Developing a (performing) workflow from reconstruction and maintenance in a database to dissemination of countrywide 3D data in an open standard.
8. Use of the 3D data in applications

3.1 Pre-processing data

To be able to use the geometries of the BAG and the BGT as input for the automatic reconstruction process, some pre-processes have been applied to make the data suitable for the 3D reconstruction process, such as correcting topological problems (not per se errors, see further), enriching the data with required additional information amd geometrically integrating the data with surrounding objects. As a result, the 2D objects are not always the same as in the original BGT and BAG datasets.

The main pre-processing operations are:

- Self-intersections are removed and arcs are discretized
- Duplicate objects are removed keeping the newest object as much as possible. Both BAG and BGT keep the history of objects. But both registers appeared to contain errors resulting in duplicate objects when selecting a specific snapshot in time. Therefore, such duplicates are detected and removed.
- Objects that touch each other (at an angle, overlap or shared boundary) are snapped and vertices are added when needed. This is done to close gaps in the BGT dataset, but also to ensure that adjacent BAG buildings actually connect.
- Other topological errors that can cause problems in the reconstruction process are restored, such as overlaps, holes, and so-called spikes. These are detected and corrected automatically. "Unclassified objects" are created for gaps in the BGT.
- Topology at different height levels is repaired. The BGT contains a planar partition at surface level (relative height

level '0'). But there are no rules to ensure topology between objects at different relative height levels that touch in space (Figure 1). This is particularly a problem for 3D reconstruction of bridges (relative height level '1') that connect to road parts at surface level (relative height level '0'), see Figure 2. Therefore, topology is repaired at those locations if possible (if gaps and overlaps are small). Where necessary, vertices on shared boundaries and intersecting objects are introduced.

- Data points are aligned on a grid for the accuracy of all validation checks. All resulting coordinates are stored with mm precision.

Figure 1: Relative height information as available in the BGT. The bridge (H) is at relative height level '1'. The bridge-parts F and G as well as the road, verge and water are at relative height level '0'. The latter form a planar partition (IMGeo, 2020)
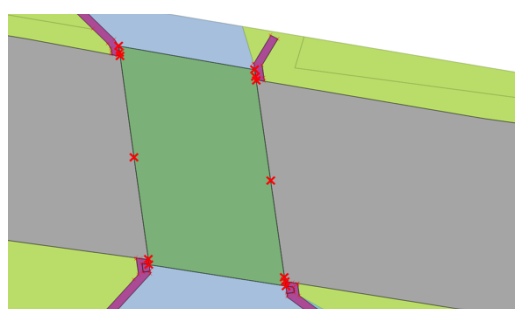

Figure 2: Bridge as modelled in BGT: The acquisition rules do not enforce topology between the green bridge at height level '1' and the grey road parts at height level '0' (as are all other objects in this example)

3.2 3dfier

For the height attribution of objects at surface level, we use the open source software 3dfier (Commandeur et al., 2019). The software automatically generates 3D city models based on 2D topography and point clouds (LAS/LAZ). The software takes 2D topographical datasets as input and "3dfies" them by lifting every polygon to 3D. The semantics of every polygon is used to perform the lifting. For example, water polygons are extruded to horizontal polygons, buildings to blocks (or to footprints depending on the parameters used), roads as smooth surfaces, etc. Every polygon is triangulated and in a next step the lifted polygons are "stitched" together so that one digital

surface model (DSM) is reconstructed (or DTM, depending on the choices of the user). The output of the software is one watertight DSM or DTM with no intersecting triangles and no holes where buildings are integrated in the surface. This surface can be used as input for urban applications, such as simulations.

3.3 Reconstructing 3D representation of buildings

For the reconstruction of building models (including the statistical information), we use the 3D BAG service described in detail in Dukai et al (2019). The service was developed to reconstruct LoD1.2 models covering the whole of the Netherlands.

Such block models can be reconstructed relatively easily from building footprints and point clouds and are widely used for example in noise or wind flow simulations. However, LoD1 representations for the same building can be different due to differences in height references and in underlying statistical calculation methods used to extrude the footprints. These differences may have an impact on the outcome of spatial analyses, although users are often not aware of these differences and their impact. To standardise possible variances of LoD1 models, and make the user aware of these variances, and to provide the option to use the appropriate height reference for a specific application, the 3D BAG service generates several reference heights per building (both for the ground surface and the extrusion height) based on different statistical values calculated on the height points that fall within the building footprint. In the 3D BAG service, the building models are generated for all ~10 million BAG buildings in The Netherlands (from AHN) and updated automatically each month. In addition, quality parameters are calculated and assigned to each building, to provide additional information on how to use the data. Table 1 provides a selection of the attributes that are calculated with the service and assigned to the buildings (Dukai et al, 2019).

Attribute	Description
ground-00, ground-10, ground-20, ground-30, ground-40, ground-50	Height of the ground surface of the building at the given percentile.
roof-25, roof-50, roof-75, roof-90, roof-95, roof-99	Height of the roof surface of the building at the given percentile. For example roof-99 is the height of the building when the roof surface is set at the 99th percentile of the z-coordinates of the point cloud of the building.
rmse-25, rmse-50, rmse-75, rmse-90, rmse-95, rmse-99	Root Mean Square Error or the geometric difference between the 3D building model and the point cloud that was used for generating the model. This measure also accounts for the whole building, not only the roof.
roof_flat	Possible values: 0: Roof is not flat 1: Roof is flat
nr_ground_pts	The number of points in the point cloud that were used for determining the ground-height of the building model. '0' means that the ground-points are missing from the point cloud at given model.
nr_roof_pts	The number of points in the point cloud that were used for determining the roof-height of the building model. '0' means that the roof-points are missing from the point cloud at given model.
height_valid	Indicates that the elevation data is actual, with respect to the building footprint.

Table 1: Selection of height and quality attributes assigned to buildings in the 3D BAG service (Dukai et al, 2019)

It should be noted that the roof_flat attribute gives a global indication about the quality of the reconstructed models, since LoD1 models representing buildings with flat roofs are most likely closer to reality.

3.4 LoD1.3

Although LoD1.2 block models serve a wide variety of applications, for some buildings the reconstructed model is not representative, for example in case of a church or a shed connected to a house.

For such cases, the LoD1.3 models (block models that represent height jumps in one building footprint) are more appropriate. It results in more realistic visualisations, but also more accurate data for simulations that take block-shaped models of buildings as input, such as noise simulation where buildings act as noise barriers.

Therefore, in a research to generate 3D input data for noise simulation, we have developed a method to generate LoD1.3 models according to the requirements of noise level calculation methods (see Figure 3). This method uses building footprints and a point cloud as input. It is described in detail in Stoter et al (2019). An improved version of that method is used in this work as well as in the next iteration of the 3D BAG service.

The LoD1.3 building reconstruction method uses the point cloud to find lines at the location of height jumps. These lines are used to subdivide the footprint polygon of the building into roofpart polygons. Each roofpart is assigned an elevation value equal to the 70th percentile of the roofpoints it contains.

Figure 3: Visualisation of reconstructed LoD1.3 buildings

The LoD1.3 reconstruction method performs for each building the following steps:
- Find the elevation points above the building footprint.
- Perform plane detection in the point cloud using a region-growing algorithm to identify all roof planes. In this step, points that are on a wall plane (facade) or not part of any plane are also removed;
- Detect the boundary of the roof planes using α-shapes and a region-growing line detection algorithm on the α-shape boundaries;
- Perform a regularisation process of the detected boundary lines. In this step lines that are close and have a very similar orientation are merged;
- Decompose the footprint polygon by inserting the regularised boundary lines into a 2D planar partition and perform a graph-cut optimisation formulation similar to Zebedin (2008) to simplify it.
- Extrude each cell in the decomposition to its representative height.

The past months, the method has been further improved and optimised so that it can reconstruct the whole of the Netherlands (~10 million buildings) within a single day on a single server. In addition, it has been integrated with the aforementioned 3D BAG service, so that the additional quality attributes are also supported for LoD1.3.

3.5 CityJSON

To disseminate the generated 3D data, we use CityJSON as an exchange format. CityJSON is an encoding of a subset of the CityGML data model (Ledoux et al, 2019), and is compacter than the GML encoding of the CityGML data model. Therefore, it is suitable for mobile and web environments. In addition, software and APIs supporting it can be easily built.

Several implementations are available to view and manipulate CityJSON files, i.e. 3D CityDB, azul (Arroyo Ohori, 2020), FME, ninja (Vitalis et al, 2020b), QGIS plugin (Vitalis et al, 2020a), val3dity (Ledoux, 2018).

The Open Geospatial Consortium (OGC) is currently considering CityJSON for adoption as an official OGC Community Standard (OGC, 2020).

3.6 Underground parts

A specific problem in the reconstruction of buildings from the BAG data is that the BAG geometry represents the outline of a building as seen from above. This BAG representation does not distinguish between parts that are above the ground and parts that are below the ground. Therefore, for buildings that have cellars extending the footprint or (part of) buildings that represent underground parking garages, the reconstructed models do not correctly represent the reality (e.g. underground parts are also extruded). To improve the reconstruction method for such buildings, we have developed a method to identify different types of multi-level buildings (Figure 4).
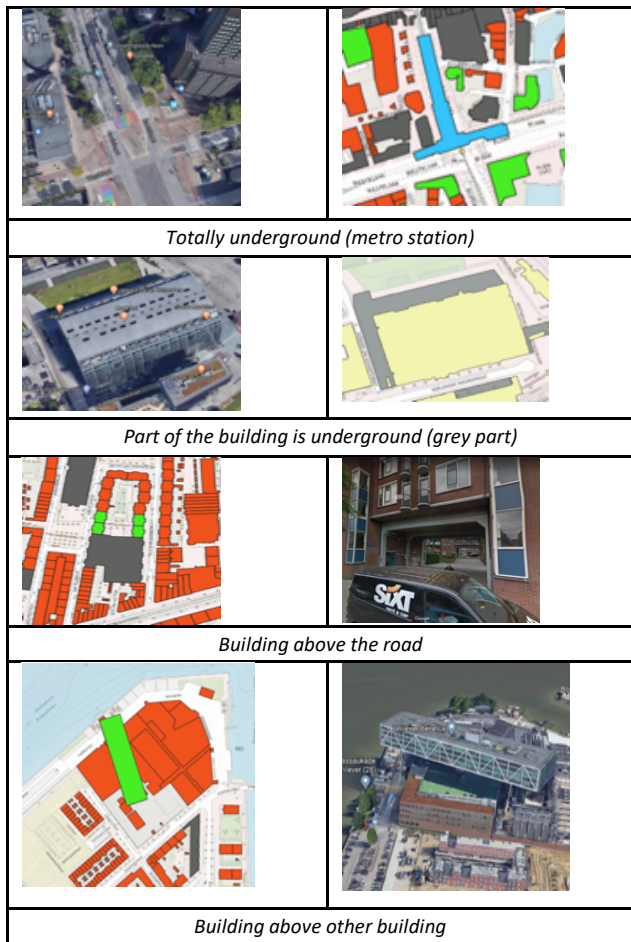

Totally underground (metro station)

Part of the building is underground (grey part)

Building above the road

Building above other building

Figure 4: Examples of multi-level building situations

The types of buildings that we distinguish and model are:
- 'normal' buildings, i.e. totally above the ground
- totally underground (e.g. metro station)
- floating building, e.g. standing on pillars on top of a road, water or another building or overhanging a road

Underground BAG buildings are identified by comparing the BAG-polygon with a polygon that is generated from the height points that represent buildings. In addition, "floating" parts of buildings (including overhangs) are identified by detecting overlaps between buildings and roads, water or other buildings. Buildings that are partially underground (e.g. a parking garage extending the footprint) are handled together with the 'normal' buildings in the reconstruction process.

3.7 Workflow from reconstruction to dissemination of countrywide 3D data

The governmental portal PDOK, that will serve the 3D data, is not 3D yet. In addition, at this moment PDOK only supports the dissemination of predefined tiles (i.e. a user cannot identify any random area of interest). The download service is therefore implemented via tiles that can be downloaded after selecting the tile(s) on an indexed map.

To make the 3D data available in these tiles, the 3D data is reconstructed from AHN tiles and BAG/BGT tiles and written into a PostGIS database. In this process, the tile boundaries are connected both in X, Y and Z so that a seamless model is built. A CityJSON writer has been developed to export the data from the database into the CityJSON tiles of size 5kmx6km. Objects that fall into multiple tiles are written into each tile in which they fall. This is done to ensure that a user does not have to search for the tile containing an object that crosses one (or more) tile boundary (boundaries). The chosen tile size is in line with BRT's (and AHN's) map sheet index so that the PDOK download service for AHN and BRT can be reused.

Deciding about the optimal tile size of these predefined tiles is finding a balance between two criteria: the smaller the tile size, the better the data can be handled by users. But the smaller the tile size, the more objects fall into multiple tiles, so the larger the data volume (when objects have to be written for more tiles). A small tile size also has the disadvantage that more objects will overlap multiple tiles.

In the future, availability on PDOK may be extended with a 3D viewer. In addition a download service may be developed to support downloads based on any area of interest. With the latter, redundant writing resulting in huge data volumes will be solved, because tiles will be generated on the fly ("on user request").

The generated 3D data is available via de PDOK download service (PDOK-3D, 2020).

3.8 Use of the 3D data in applications

The final goal of this 3D data project, is to support applications with standardised and future-proof 3D topographic data. At this moment the 3D data can directly be used in 3D noise simulation software as prescribed by the Dutch government (Stoter et al, 2020; Kumar et al, 2020). Other applications are also foreseen. By providing a first version of a countrywide 3D dataset, we aim to collect further users' feedback as well as further requirements for the data which will be used to improve the next version of the data.

4. RESULTS AND CONCLUSIONS

In this paper we have presented the workflow that we are currently developing to generate and disseminate a 3D large scale topographic dataset covering the whole of the Netherlands. The Kadaster is currently implementing the workflow to generate the 3D model of the Netherlands containing the three different data sets that were described in section 2, and make it available in the CityJSON dataformat. In addition, the workflow will be re-run every year with updated input data.

The first product (BGT surfaces with LoD1.2 buildings) has recently been published via the governmental portal (PDOK-3D, 2020). See figure 5.

Figure 5: Visualisation of 3D Basisbestand Volledig as available from PDOK

Later this year (2020) the other two products - *3D Basisbestand Gebouwen* and *3D Hoogtestatistieken Gebouwen*- will follow.

Based on feedback, we will improve the several parts of the workflow and finetune parameters that we use in the different steps. We will also study new applications that require 3D data and adjust the data if needed.

In the future, we will also investigate how we can make best use of updated point clouds and specifically the point cloud that is generated each year from dense image matching by Kadaster. In this, we will also investigate what the impact is on the reconstructed data when using different point clouds each time the data is reconstructed. As part of this, we will study the best ways to generate and maintain one integrated height reference model of the Netherlands combining height data acquired by different acquisition techniques and different organisations.

Another field of further study is the integration and alignment of 3D city models of other governments (like large cities) to provide consistent 3D data that is produced by governments.

Future work will also focus on generating one aggregated quality value per building. In the current version there are several attributes assigned to building models that express the quality of the building. But it needs data-expertise to interpret the values in a correct way. Therefore, as requested by domain-users, we plan to aggregate the values into one overall quality assessment for each building. This aggregated value per building will be based on a statistical analysis of all values in the whole dataset.

Finally, to make the data easier accessible we plan to develop a 3D viewer and a download service that is capable of providing data based on any area defined by the user.

ACKNOWLEDGEMENTS

REFERENCES

AHN, 2020, https://www.ahn.nl

Arroyo Ohori, K., 2020. Azul: a fast and efficient 3D city model viewer for macOS.. Transactions in GIS, 2020. ISSN: 1467–9671. https://doi.org/10.1111/tgis.12673

Digital Government, 2020, Digitale Overheid. https://www.digitaleoverheid.nl/dossiers/basisregistraties/

Biljecki, F., H. Ledoux, and J. Stoter, 2016. An improved LOD specification for 3D building models. Computers, Environment and Urban Systems, 59: 25–37. https://doi.org/10.1016/j.compenvurbsys.2016.04.005

Brink, L. van den, J. Stoter, S. Zlatanova, 2013. UML-Based approach to developing a CityGML Application Domain Extension, Transactions in GIS 940 17 (6) (2013) 920–942. https://doi.org/10.1111/tgis.12026

Brink, L. van den, J. Stoter, S. Zlatanova, 2012. Establishing a national standard for 3D topographic data compliant to CityGML, International Journal of Geographical Information Science 27 (1) (2013) 92–113 https://doi.org/10.1080/13658816.2012.667105

Commandeur, T., H. Ledoux, R. Peters, M. Pronk, B. Dukai, F. Biljecki, A. Labetski, 2019. tudelft3d/3dfier: version 1.2 (August 2019). doi.org/10.5281/zenodo.3378858

Dukai, B., H. Ledoux and J. Stoter, 2019. A Multi-Height LoD1 Model of all Buildings in the Netherlands. 14th 3D GeoInfo Conference 2019, ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences IV-4(W8), ISPRS, 2019, pp. 51–57. https://doi.org/10.5194/isprs-annals-iv-4-w8-51-2019

IMGeo, 2020. Objectenhandboek BGT | IMGeo, https://imgeo.geostandaarden.nl/def/imgeo-object/overbruggingsdeel/overbruggingsdeel

Kumar, K., H. Ledoux, R. Schmidt, T. Verheij and J. Stoter, 2020. ISPRS International Journal of Geo-Information 9(2), 2020, pp. 121. https://doi.org/10.3390/ijgi9020121

Ledoux, H. K. Arroyo Ohori, K. Kumar, B. Dukai, A. Labetski and S. Vitalis, 2019. CityJSON: a compact and easy-to-use encoding of the CityGML data model. Open Geospatial Data, Software and Standards 4 (4), 2019. https://doi.org/10.1186/s40965-019-0064-0

Ledoux, H. 2018. Val3dity: validation of 3D GIS primitives according to the international standards Open Geospatial Data, Software and Standards 3 (1), 2018, pp. 1. https://doi.org/10.1186/s40965-018-0043-x

OGC, 2020.https://www.ogc.org/pressroom/pressreleases/3152

Oude Elberink, S., J. Stoter, H. Ledoux and T. Commandeur, 2-13. Generation and dissemination of a national virtual 3D city and landscape model for the Netherlands. Photogrammetric Engineering & Remote Sensing 79(2), 2013, pp. 147–158.

PDOK-3D, 2020. https://brt.kadaster.nl/basisvoorziening-3d/

Stoter, J. H. Ledoux, M. Reuvers, L. van den Brink, R. Klooster, P. Janssen, J. Beetz, F. Penninga and G. Vosselman, 2013. Establishing and implementing a national 3D standard in the Netherlands. Photogrammetrie, Fernerkundung, Geoinformation 4, 2013, pp. 381–392. https://doi.org/10.1127/1432-8364/2013/0184

Stoter, J., R. Peters, T. Commandeur, B. Dukai, K. Kumar and H. Ledoux, 2020. Automated reconstruction of 3D input data for noise simulation. Computers, Environment and Urban Systems 80, 2020, pp. 101424. https://doi.org/10.1016/j.compenvurbsys.2019.101424

Stoter, J., C. Roensdorf, R. Home, D. Capstick, A. Streilein, T. Kellenberger, E. Bayers, P. Kane, J. Dorsch, P. Wo'zniak, G. Lysell, T. Lithen, B. Bucher, N. Paparoditis and R. Ilves, 2014. 3D modelling with national coverage bridging the gap between research and practice 9th International 3DGeoInfo 2014, November 2014 https://doi.org/10.1007/978-3-319-12181-9_13

Stoter, J., B. Vallet, T. Lithen, M. Pla, P. Wozniak, T. Kellenberger and H. Ledoux, 2016. State-of-the-art of 3D national mapping in 2016. Int. Arch. Photogramm. Remote Sens. Spatial Inf. Sci., XLI-B4, 2016, Prague, 2016. https://doi.org/10.5194/isprs-archives-XLI-B4-653-2016

Vitalis, S., K. Ohori and J. Stoter, 2020a. CityJSON in QGIS: Development of an open-source plugin Transactions in GIS, June 2020. ISSN: 1467-9671 https://doi.org/10.1111/tgis.12657

Vitalis, S., A. Labetski, X. Li, F. Dahle, Fr. Boersma, K. Arroyo Ohori, H. Ledoux and J. Stoter, 2020b. CityJSON + Web = ninja. 3DGeoInfo 2020 conference/BIM GIS Workshop, September 2020, organised by UCL and Ordnance Survey

Zebedin, L., Bauer, Joachim and Karner, Konrad and Bischof, Horst, 2008. Fusion of feature-and area-based information for urban buildings modeling from aerial imagery. European conference on computer vision. 2008, pp. 873-886. https://doi.org/10.1007/978-3-540-88693-8_64